

**UNIVERSIDADE FEDERAL DE ALAGOAS
CENTRO DE TECNOLOGIA
PROGRAMA DE PÓS-GRADUAÇÃO
EM ENGENHARIA CIVIL - ESTRUTURAS**

THABATTA MOREIRA ALVES DE ARAÚJO

**REDES NEURAS INCORPORADAS A MÉTODOS NUMÉRICOS PARA SOLUÇÃO DE
PROBLEMAS DE ENGENHARIA ESTRUTURAL**

MACEIÓ/AL

2015

THABATTA MOREIRA ALVES DE ARAÚJO

**REDES NEURAIIS INCORPORADAS A MÉTODOS NUMÉRICOS PARA SOLUÇÃO DE
PROBLEMAS DE ENGENHARIA ESTRUTURAL**

Dissertação de mestrado apresentada ao Programa de Pós-Graduação em Engenharia Civil da Universidade Federal de Alagoas como requisito parcial para obtenção do título de Mestre em Engenharia Civil.

Orientador: Professor Eduardo Toledo de Lima Junior

Co-orientador: Professor João Carlos Cordeiro Barbirato

MACEIÓ/AL

2015

Catálogo na fonte
Universidade Federal de Alagoas
Biblioteca Central
Divisão de Tratamento Técnico
Bibliotecário Responsável: Valter dos Santos Andrade

A663r Araújo, Thabatta Moreira Alves de.
Redes neurais incorporadas a métodos numéricos para solução de problemas de engenharia estrutural / Thabatta Moreira Alves de Araújo. – Maceió, 2015.
89 f. : il., graf. e tabs.

Orientador: Eduardo Toledo de Lima Junior.
Coorientador: João Carlos Cordeiro Barbirato.
Dissertação (Mestrado em Engenharia Civil) – Universidade Federal de Alagoas. Centro de Tecnologia. Programa de Pós-Graduação em Engenharia Civil. Maceió, 2015.

Bibliografia: f. 85-89.

1. Redes neurais artificiais. 2. Métodos dos elementos finitos. 2. Interpolação.
3. Extrapolação. I. Título.

CDU: 624.04



Universidade Federal de Alagoas – UFAL
Unidade Acadêmica Centro de Tecnologia – CTEC
Programa de Pós-Graduação em Engenharia Civil – PPGEC



**REDES NEURAIS INCORPORADAS A MÉTODOS NUMÉRICOS PARA
SOLUÇÃO DE PROBLEMAS DE ENGENHARIA ESTRUTURAL**

THABATTA MOREIRA ALVES DE ARAÚJO

Dissertação submetida à banca examinadora do Programa de Pós-Graduação em Engenharia Civil da Universidade Federal de Alagoas e aprovada no dia 02 do mês de outubro do ano de 2015.

Banca Examinadora:

Eduardo Toledo de Lima Jr

Prof. Dr. Eduardo Toledo de Lima Junior
Orientador – PPGEC/CTEC/UFAL

João Carlos Cordeiro Barbato

Prof. Dr. João Carlos Cordeiro Barbato
Coorientador – PPGEC/CTEC/UFAL

Francisco Patrick Araújo Almeida

Prof. Dr. Francisco Patrick Araújo Almeida
PPGEC/CTEC/UFAL

Pyramo Pires da Costa Júnior

Prof. Dr. Pyramo Pires da Costa Júnior
PUC-MG



Campus A. C. Simões, Av. Lourival de Melo Mota, S/N
Tabuleiro do Martins – CEP 57072-970 – Maceió – Alagoas
Tel/Fax: (82) 3214-1863
E-mail: ppgec@ctec.ufal.br
Homepage: www.ctec.ufal.br/posgraduacao/ppgec



Dedico aos que não ofuscaram seus objetivos e o alcançaram com muito custo.

AGRADECIMENTOS

Agradeço a Deus.

Agradecer a Ele é como agradecer a todos, pois Ele está em tudo que me cerca: família, amigos, parentes, colegas. Em especial aos meus queridos orientadores Eduardo e João, pela paciência e amizade nesse tempo.

À CAPES pelo fomento aos estudos.

Enfim, a todos aqueles que tornaram esses anos de estudo mais proveitosos e felizes.

RESUMO

O desenvolvimento de novos materiais e tecnologias, associado a projetos estruturais cada vez mais complexos, demandam análises numéricas robustas e precisas a um custo computacional adequado. Estudos em modelagem computacional têm evoluído no sentido de suprir a essa demanda aprimorando continuamente os métodos numéricos já existentes, bem como propondo novas técnicas. Dentre os métodos amplamente utilizados, o Método dos Elementos Finitos (MEF) tornou-se uma potente ferramenta de busca por soluções no âmbito da engenharia estrutural. Também surgiram abordagens alternativas de resolução relacionadas à modelagem, tais como as técnicas de inteligência artificial. Uma dessas técnicas, as Redes Neurais Artificiais (RNA), são redes projetadas para modelar capacidades humanas como aprendizagem e generalização através de estrutura de processamento que utiliza o conexionismo como paradigma. As RNA têm sido aplicadas nas mais diversas áreas na solução de uma gama de problemas que envolvem classificação, aproximação de funções, previsão, dentre outros. Este trabalho explora maneiras de incorporar RNA a métodos aproximativos tradicionais como o Método dos Elementos Finitos para aplicação em problemas de elasticidade plana, considerando não linearidade geométrica.

Palavras-chave: Redes Neurais Artificiais. MEF. Interpolação. Extrapolação.

ABSTRACT

The development of new materials and technologies, associated to increasingly complex problems in structural engineering, require more robust and accurate numerical analyses with an affordable computational cost. Computer modeling has been evolving by improving existing numerical methods and also developing new ones. Among the methods widely applied, finite elements have become a powerful tool in the search for solutions in structural engineering. Other approaches also emerged as alternative modeling strategies, such as artificial intelligence techniques. One of these techniques, Artificial Neural Networks (ANN), is designed for simulating human connexionism paradigm. ANN has been applied in several areas to solve a range of problems concerning classification, function fitting, forecasting, among others. This work explores ways to coupling ANN to traditional approximation methods such as the Finite Element Method (FEM) for applications in plane elastostatics, considering geometrically nonlinear aspects.

Keywords: Artificial Neural Network. FEM. Interpolation. Extrapolation.

LISTA DE ABREVIATURAS E SIGLAS

ADALAINE - *ADaptive LINear Element*

ANN – *Artificial Neural Network*

FEM – *Finite Element Method*

LMS - *least mean square rule*

MATLAB – *Matrix Laboratory*

MEF – Método dos Elementos Finitos

MLP – *Multi Layer Perceptron*

PCA – Análise de Componentes Principais

Q4- Elemento finito quadrilateral de quatro nós

Q9- Elemento finito quadrilateral de nove nós

RNA- Redes Neurais Artificiais

SUMÁRIO

1	INTRODUÇÃO	10
1.1	Considerações Iniciais	10
1.2	Motivação	11
1.3	Objetivos.....	12
1.4	Estrutura da Dissertação	12
2	REDES NEURAI APLICADAS EM ENGENHARIA DE ESTRUTURAS	13
3	FORMULAÇÃO DO MÉTODO DOS ELEMENTOS FINITOS.....	18
3.1	Considerações Iniciais	18
3.2	Funções Aproximativas e Sistemas de Coordenadas	20
3.3	Tipos de Elementos Finitos	21
3.4	Método dos Elementos Finitos.....	22
3.5	Formulação em Deslocamentos do MEF aplicado à Elasticidade Plana	24
3.5.1	Formulação bidimensional linear do Método dos Elementos Finitos	24
3.5.2	Formulação bidimensional não linear geométrica do Método dos Elementos Finitos.....	27
4	REDES NEURAI ARTIFICIAIS	33
4.1	Conceitos	33
4.2	Neurônio Biológico	33
4.3	Neurônio Artificial	34
4.4	Breve Histórico.....	37
4.5	Arquitetura de Redes Neurais.....	38
4.6	Aprendizagem.....	39
4.6.1	Aprendizado não supervisionado.....	39
4.6.2	Aprendizado supervisionado	40
4.7	Redes de Camadas Múltiplas (MLP – <i>Multilayer Perceptron</i>).....	40
4.8	Base Radial	41
4.9	Algoritmo de Retro Propagação (<i>Backpropagation</i>).....	41

4.10	Treinamento Rápido	45
4.10.1	Método de Levenberg-Marquardt.....	45
4.11	Eficiência de uma Rede Neural	46
5	TÉCNICA PROPOSTA APLICADA A ANÁLISE DE ESTRUTURAS	49
5.1	Metodologia.....	49
5.2	Problemas com Solução Analítica Conhecida	51
5.2.1	Viga em balanço sob carga pontual	51
5.2.2	Modelo constitutivo de Popovics (1973)	56
5.3	Problemas com Solução Aproximada via Formulação Linear do MEF	62
5.3.1	Chapa sujeita a carga pontual	63
5.3.2	Modelo simplificado de barragem em contraforte	66
5.4	Problemas com Solução Aproximada via Formulação Não Linear do MEF	73
5.4.1	Viga em balanço sob carregamento distribuído	73
5.4.2	Coluna com carga excêntrica.....	77
6	CONSIDERAÇÕES FINAIS.....	82
	REFERÊNCIAS BIBLIOGRÁFICAS	85

1 INTRODUÇÃO

1.1 Considerações Iniciais

A modelagem numérica surgiu com a necessidade de técnicas de análise e compreensão de fenômenos cuja complexidade era restrita às soluções analíticas. Neste contexto surgiram os métodos aproximativos de resolução objetivando tratar a complexidade contida no fenômeno em tempo relativamente menor. Alguns dos métodos estudados na modelagem computacional com direcionamento à solução de problemas típicos das engenharias são: Métodos dos Elementos Finitos, Métodos dos Elementos de Contorno, Método dos Volumes Finitos, Métodos das Diferenças Finitas, dentre outros. Esses métodos já são consolidados no meio científico e são amplamente utilizados na modelagem de áreas tão diversas como a mecânica do contínuo, nanotecnologia, bioinformática, entre outros. Tais métodos traduzem computacionalmente conceitos físicos e matemáticos através de formulações que contemplam sistemas de equações, no intuito de desenvolver soluções adequadas ao problema científico abordado.

Paralelamente à evolução das análises numéricas surgiram técnicas computacionais alternativas de resolução de problemas os quais a modelagem numérica convencional até então era custosa. Essa abordagem alternativa ficou conhecida como inteligência artificial e abrange uma série de técnicas computacionais tais como as Redes Neurais Artificiais, os sistemas Fuzzy, os Algoritmos Genéticos, entre outras. Assim como os métodos numéricos, essas técnicas tornaram-se atrativas para solucionar problemas, no entanto, com focos diferentes por atenderem um campo onde a fenomenologia é descrita por modelos chamados caixa preta. Por caixa preta são conhecidos os sistemas potencialmente complexos, no qual as análises limitam-se ao mapeamento de entradas e saídas, já que a estrutura interna apresenta-se desconhecida ou não levada em consideração. Dentre as técnicas alternativas já mencionadas as Redes Neurais Artificiais consolidaram-se como pioneiras na modelagem “inteligente”. De abordagem conexionista, também denominada de biológica, de ênfase no modelo de funcionamento do cérebro, dos neurônios e das conexões neurais, as RNA apresentaram-se como

emulador das habilidades humanas para *softwares* e *hardwares*. Assim tornaram-se uma ferramenta interessante de análise de acordo com Hopfield (1982), Rumelhart *et al.* (1986), Cybenko (1988), e outros, que provaram que as RNA atendem a um grande contingente de casos e fornecem soluções confiáveis.

1.2 Motivação

Assim como no cérebro humano os neurônios artificiais aprendem por experiência e associação. Desta forma, em posse de um conjunto de par de dados, constituído pelas entradas e saídas correspondentes, a estrutura interna da rede é ajustada de forma a generalizar a solução para dados não contidos no domínio do ajuste. Este ajuste é chamado de aprendizado da rede e permite que sejam mapeadas entradas e saídas de forma a aproximar funções matemáticas das mais diversas, lineares ou não. Por isso Haykin (1999) descreve as Redes Neurais Artificiais como aproximadores universais. Assim, uma vez que a estrutura interna está ajustada a RNA demonstra sua capacidade de generalização através de resposta em simulação, para qualquer entrada, a tempos relativamente baixos se comparados a outras técnicas para a modelagem de um mesmo problema. Portanto, esse aprendizado confere às RNA versatilidade e vantagens em demandas difíceis e restritivas tornando a resolução simplificada a um custo menor. No entanto, na prática, Redes Neurais Artificiais funcionam como ferramenta no processo de obtenção de solução. O projeto das redes deve associar fundamentação teórica e conhecimento acerca do problema a ser modelado. Este deve ser enquadrado em um dos paradigmas da rede tais como reconhecimento de padrões, memória associativa, previsão, entre outras. Objetivando expansão das aplicações as RNA evoluíram e têm sido combinadas a métodos numéricos. Essa ampliação no horizonte de aplicações possibilitou a modelagem de problemas de forma totalmente computacional, o que tem mudado a forma como muitos problemas são normalmente tratados e solucionados.

Portanto, entende-se que a combinação de Redes Neurais com métodos numéricos pode incorrer em melhora no desempenho da execução dos códigos computacionais de tais métodos. As redes neurais podem emular o método numérico, para fornecer respostas para quaisquer possibilidades de valores de entrada, criando superfícies de resposta que associem as variáveis que regem um determinado problema. Assim pode-se reduzir a quantidade de simulações em métodos numéricos, além de relacionar variáveis

como número de elementos e tempo de simulação, guardados os níveis de precisão desejados para a análise. Desse modo, a modelagem com RNA pode tornar-se uma alternativa atraente à novas simulações com métodos numéricos convencionais.

1.3 Objetivos

O objetivo principal deste trabalho é desenvolver e programar códigos inteligentes integrados a um dos métodos numéricos mais difundidos em análise estrutural, através da junção de Redes Neurais Artificiais (RNA) ao Método dos Elementos Finitos (MEF).

Desta forma, com a integração da RNA aos mecanismos de solução de problemas de engenharia de estruturas espera-se estabelecer metodologias alternativas de resolução para ampliar as possibilidades de estudo e análise de casos. São abordados problemas unidimensionais e planos de elasticidade a fim de validar a metodologia proposta e ilustrar sua eficácia, justificando posteriores aplicações em problemas de maior complexidade e dimensão.

1.4 Estrutura da Dissertação

O trabalho está subdividido em seis capítulos incluindo a Introdução e as Considerações Finais.

O primeiro Capítulo é a presente Introdução. No Capítulo 2 listam-se alguns trabalhos que relacionam redes neurais à métodos numéricos.

No Capítulo 3 encontram-se os fundamentos teóricos acerca das formulações a serem utilizadas em MEF linear e não linear geométrico.

No Capítulo 4 são apresentados alguns fundamentos teóricos sobre as Redes Neurais Artificiais.

No Capítulo 5 apresenta-se a metodologia proposta, bem como exemplos de aplicação numérica em problemas clássicos de engenharia estrutural.

No Capítulo 6 apresentam-se as considerações finais deste trabalho, as dificuldades encontradas, e sugestões para possíveis desenvolvimentos futuros.

2 REDES NEURAI APLICADAS EM ENGENHARIA DE ESTRUTURAS

A modelagem computacional tem evoluído para atender o crescente grau de complexidade das análises no campo da engenharia estrutural. Como consequência, o modo como se tem tratado os problemas também evoluiu. Surgiram procedimentos híbridos, que mesclam métodos numéricos a algoritmos de inteligência artificial como as Redes Neurais, Algoritmos Genéticos, Fuzzy entre outros, no intuito de promover alternativas de resolução aos métodos convencionais. Esse tipo de abordagem alternativa permitiu que muitos problemas pudessem ser modelados e solucionados.

Quando surgiram, logo após a segunda Guerra Mundial, as Redes Neurais Artificiais restringiam sua aplicação, em grande parte, a problemas de eletrônica e computação. Com o passar dos anos, a evolução dos algoritmos e dos computadores permitiu a expansão no campo de aplicação das RNA. Na literatura os primeiros trabalhos encontrados contendo Redes Neurais Artificiais em conjunto com Engenharia de Estruturas aparecem no início dos anos 90 embalados, em grande parte, por problemas de geomecânica. No entanto, atualmente se encontram referências nas mais diversas áreas da engenharia de estruturas, tais como análises estruturais estáticas e dinâmicas, identificação, predição, danos e modelos de comportamentos de materiais, entre outros.

Chen *et al.* (1995) utilizaram Redes Neurais para controlar a intensidade de aplicação de carga dinâmica sobre estruturas. O trabalho é dividido em duas partes. A primeira consiste na representação da estrutura a ser controlada através de um emulador e a segunda parte consiste no controle da ação do carregamento sob a estrutura. A arquitetura da rede projetada era constituída de 4 neurônios de entrada, 5 neurônios na camada intermediária e 1 neurônio na saída. As entradas consideradas foram os deslocamentos ao longo da estrutura, e a saída o deslocamento no próximo passo de carregamento.

Abdallah e Stavroulakis (1995) utilizaram redes neurais treinadas com dados experimentais de conexões metálicas para estimar o comportamento mecânico de

conexões metálicas semirrígidas. Foram considerados dois tipos de estruturas de conexões.

Ghaboussi *et al.* (1991) treinaram uma rede *Multilayer Perceptron* (MLP), a partir de dados experimentais, para modelar o comportamento do concreto em estado plano de tensão sob carregamento monotônico biaxial e compressão uniaxial cíclica. As tensões e deformações e seus incrementais foram utilizadas como variáveis de entrada. A camada intermediária era composta por 6 neurônios. A saída foi constituída de deformações em duas direções.

Kirkegaard e Rytter (1993) aplicaram uma rede *Multilayer Perceptron* na avaliação de danos em vigas com base na medida de abertura de fraturas em vigas de aço submetidas à vibração. A rede foi treinada com 5 modos de frequência. A arquitetura foi 5:7:7:2, onde os neurônios de saída eram as variáveis de localização na geometria e tamanho da fratura.

Stephens e Van Luchene (1994) exploraram o uso de RNA para ponderar o dano em determinada condição de segurança da estrutura submetida a eventos sísmicos. A entrada da rede neural consistia em três tipos de danos: máximo deslocamento, energia dissipada acumulada e degradação na rigidez. A rede foi modelada com uma camada intermediária de 7 neurônios, utilizando um programa de computador, o NNICE.

Flood e Kartam (1994) introduziram conceitos, limitações teóricas e eficiência de Redes Neurais na análise de estruturas simples. São discutidas questões como número de camadas e neurônios, distribuição e formato dos padrões de treinamento, validação da rede e velocidade de processamento dos algoritmos. Como conclusão, tem-se que o sucesso de aplicabilidade das Redes Neurais depende de fatores como qualidade dos dados utilizados para treinamento, arquitetura da rede, método de treinamento e o modo como os dados de entrada e saída são interpretados.

Goh *et al.* (1995) estimaram os deslocamentos máximos em um trecho de escavação em argilas através do treinamento de uma MLP com o algoritmo *backpropagation* e padrões obtidos por modelagem em MEF. Os dados foram gerados por elementos finitos paramétricos, e a topologia de rede adotada foi de 7:3:1. As variáveis de treinamento foram: largura e altura da superfície escavada, profundidade do solo, tensões no contorno e rigidez do solo. A saída foi a deflexão máxima da parede lateral

Topping e Bahreininejad (1995) estudaram o uso de redes recorrentes de aprendizado não supervisionadas na criação de malhas adaptativas não estruturadas de elementos finitos para análises dinâmicas. A abordagem dividiu grosseiramente o domínio em um número previsto de elementos triangulares (T800), e seguidamente, dentro de cada elemento era feita uma nova divisão da geometria. O procedimento era realizado a cada passo de carregamento.

Khan *et al.* (1993) desenvolveram Redes Neurais para geração de elementos em subdomínios de elementos finitos. Para acelerar o *backpropagation* foi utilizado um método de programação paralela. Os dados de entrada para a rede foram dados em relação à geometria dos elementos individuais e os parâmetros de malha nodais. A saída da rede foi o número de elementos finitos triangulares e nós gerados para cada malha. A rede neural foi desenvolvida utilizando um programa específico NETS 2.01.

Begum *et al.* (1995) desenvolveram uma Rede Neural Artificial para um sistema de testes de predição de fratura ao longo de uma superfície de concreto. Um modelo em MEF foi utilizado para simular componentes de fratura sujeitos a carregamento de impacto. A entrada para a rede foi a amplitude dos pontos de amostragem equidistantes sobre o espectro de amplitude-frequência. A saída da rede foi a probabilidade de defeitos que ocorrem dentro de um dado intervalo de profundidade.

Lefik (1997) propôs um modelo utilizando MEF e RNA para descrever leis constitutivas de materiais compósitos como substituição à abordagem fenomenológica ou puramente numérica. Foram apresentados alguns exemplos para os quais Lefik utiliza modelos em elementos finitos específicos para obtenção de dados para treinamento. Uma rede recorrente é inicialmente treinada com dados MEF para obtenção de pares de deformação e tensão em determinados pontos de Gauss sob alguns níveis de carregamento. Os deslocamentos em pontos estratégicos foram monitorados ao longo da estrutura e posteriormente comparados com dados conhecidos que descrevem o comportamento do compósito.

Bakhary *et al.* (2007) estudaram como detectar o dano em estruturas submetidas à vibração utilizando dados de modelos em Elementos Finitos para treinamento de Redes Neurais MLP. O primeiro estágio consiste de uma rede treinada tendo como entradas as frequências e modos de vibração de estrutura não discretizada e como saída as frequências na estrutura já discretizada. Já o segundo estágio recebe como entrada a saída

do primeiro estágio e o modo de vibração na estrutura discretizada. Como saída obteve-se a localização do dano na estrutura. A metodologia é ilustrada por meio de exemplos.

Bağcı e Altintas (2006) realizam uma análise de estado plano utilizando RNA em estruturas de geometria quadrada, reforçadas e confinadas de concreto. Os dados para a rede foram obtidos através de uma análise estática inelástica de elementos finitos adotando procedimento incremental e iterativo, considerando não linearidade dos materiais e carregamento incremental sob a face lateral. O modelo considera o esmagamento da cobertura e do núcleo de concreto, o encruamento do aço e os efeitos do confinamento. Os resultados obtidos foram comparados com dados de simulações puramente em MEF e dados realísticos.

Lefik (2013) tratou de aspectos relativos à aplicação de Redes Neurais à modelagem numérica em engenharia civil com foco em relações constitutivas geradas a partir de modelos, aos quais denominam híbridos, entre MEF e RNA. É proposto inicialmente treinar uma rede neural com dados MEF para estimar grandezas que em uma próxima estimativa de passo de carregamento são incorporadas aos dados de treinamento. Assim, a grandeza encontrada a cada incremento foi utilizada como realimentação da rede neural, de arquitetura recorrente, com pesos reajustados a cada passo de carregamento.

Castro (2009) desenvolveu códigos de redes neurais e algoritmos genéticos para a otimização de estruturas reticuladas e contínuas 2D e 3D. A utilização dos algoritmos genéticos visam obter a solução para um problema específico usando uma estrutura de dados baseada em um cromossomo, uma das possíveis soluções do problema. Utilizou de métodos como elementos finitos, elementos de contorno, método das soluções fundamentais e método da reciprocidade dual combinados a algoritmos genéticos. A utilização das Redes Neurais visa generalizar o processo de otimização em substituição aos métodos clássicos de análise estrutural, que normalmente são feitos a partir da minimização da função objetivo. São apresentados vários exemplos para demonstrar a metodologia do trabalho.

Almeida (2012) utilizou Redes Neurais para aproximar o comportamento de estruturas não lineares. Em uma primeira fase, desenvolveu códigos em MATLAB® para resolver problemas unidimensionais e testar a capacidade da rede. Em outra fase, utilizou de análise de uma treliça de seis barras e a estrutura de um casco de navio.

Lopes (2007) utiliza redes neurais para identificar danos estruturais através da modelagem não linear das estruturas. Uma rede neural é treinada com pares de dados de potenciais internos na estrutura e parâmetros de dano advindos de simulação em método dos Elementos de Contorno. Os resultados fornecidos por RNA são comparados a resultados obtidos via algoritmos genéticos.

3 FORMULAÇÃO DO MÉTODO DOS ELEMENTOS FINITOS

3.1 Considerações Iniciais

Em várias áreas do conhecimento, inclusive em mecânica estrutural, existem problemas cuja solução analítica é desconhecida ou de difícil desenvolvimento, para os quais pode-se obter respostas aproximadas, com o uso de métodos numéricos. Tais métodos propõem o cálculo das grandezas de interesse em pontos discretos do domínio em análise, como uma aproximação para a solução do problema.

O modo como os algoritmos dos métodos numéricos são formulados e aplicados depende, em parte, das características inerentes a cada caso. São problemas de eletromagnetismo, mecânica das estruturas, bioengenharia e mecânica dos fluidos, entre outros fenômenos. Lima Junior (2006) pondera que a análise de uma estrutura em um determinado regime deve ser pautada em algumas hipóteses a respeito do seu material constituinte, sua geometria e condições de serviço. Para a mecânica das estruturas essas hipóteses são baseadas na teoria da elasticidade. Esta teoria estabelece as equações diferenciais que regem diferentes situações da mecânica do contínuo e explana a respeito do estado de tensões e deformações de um corpo de modo a criar hipóteses e considerações utilizadas posteriormente em formulações numéricas. Assim, a partir dos resultados desejados e da complexidade do fenômeno é possível escolher ferramentas mais apropriadas para cada tipo de abordagem.

A priori a classificação dos problemas auxilia na escolha do método mais adequado a ser aplicado de forma a não comprometer o desempenho e precisão de resultados. Grosseiramente, problemas em mecânica das estruturas podem ser divididos quanto:

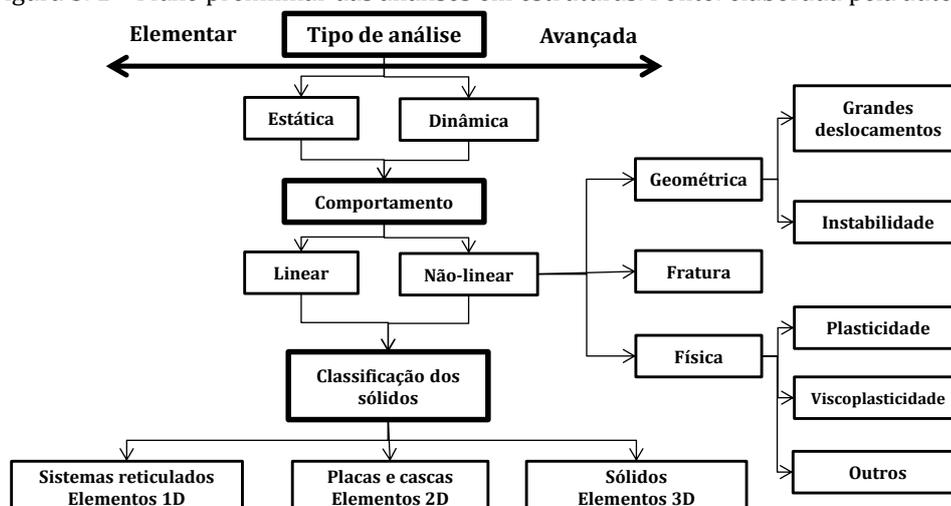
- À natureza: análises dinâmicas consideram a inércia associada às acelerações a que cada um dos componentes é sujeito. Entretanto, quando as ações são aplicadas de um modo suficientemente lento, as forças de inércia tornam-se desprezíveis, admitindo-se portanto, a análise como estática.
- Ao comportamento: quando os deslocamentos provocados pelas ações externas

são muito pequenos em comparação às dimensões dos componentes da estrutura, considera-se a estrutura em regime de pequenos deslocamentos e deformações. Admite-se, portanto, que não existe influência da modificação da geometria da estrutura na distribuição dos esforços e das tensões e o estudo é feito com base na geometria inicial indeformada. Se esta hipótese não for considerada, a análise é designada geometricamente não linear. É necessário considerar, também, o material que constitui a estrutura, se a relação entre tensões e deformações do material é linear. Nos casos onde se deve considerar a não linearidade do modelo constitutivo do material recorre-se a formulações específicas de análise não linear, sendo portanto, o comportamento não linear físico.

- À geometria: as estruturas podem ser classificadas quanto à sua geometria como reticuladas, laminares ou sólidas. Estas últimas são as mais genéricas, sendo classificadas como sólidas as que não apresentarem características que as permitam enquadrar no grupo das laminares ou das reticuladas. As estruturas laminares são aquelas cuja espessura é muito inferior ao restante das outras dimensões. Já as estruturas reticuladas são as constituídas por barras, cujas dimensões transversais são muito menores do que o comprimento do respectivo eixo.

A Figura 3. 1 traz um fluxo preliminar mais usual para a análise de casos que atendam a engenharia de estruturas.

Figura 3. 1 – Fluxo preliminar das análises em estruturas. Fonte: elaborada pela autora.



Dentre os métodos numéricos baseados na modelagem das equações diferenciais do problema, destaca-se o Método dos Elementos Finitos (MEF), o qual será brevemente

descrito neste capítulo.

3.2 Funções Aproximativas e Sistemas de Coordenadas

Um dos modos de construir a aproximação pelo MEF é utilizar formulações que envolvam funções de aproximação. Essas funções são construídas pela subdivisão do domínio em elementos e pela construção de funções intrínsecas associadas a cada elemento. Tais funções devem atender às condições de continuidade e completude para que a convergência do método seja garantida.

Por continuidade entende-se que as funções de aproximação devem ser suficientemente suaves. O grau de suavidade depende da ordem das derivadas que aparecem na equação diferencial que rege o problema. Por completude entende-se o polinômio que possui todos os termos completos até o seu mais alto grau.

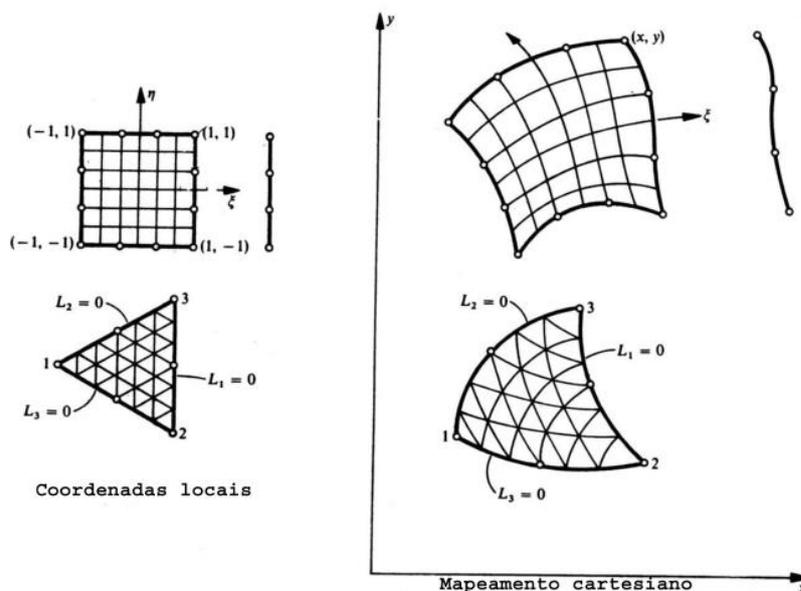
A teoria da elasticidade exige que os campos de deslocamentos possam assumir valores constantes para que os elementos finitos possam representar exatamente movimento de corpo rígido e os estados de deformação constante. Assim, à medida que os tamanhos dos elementos se aproximam de zero, as funções de aproximação (e suas derivadas, incluindo até a derivada de ordem superior que aparece na formulação) devem ser capazes de assumir valores constantes. Isso fica garantido a partir da condição de completude, a capacidade que um conjunto de funções tem de se aproximar de uma dada função suave com certo grau de precisão.

Existem situações onde a substituição do sistema de coordenadas cartesianas por outro sistema faz-se adequado devido as características geométricas de determinados elementos. O propósito da utilização do sistema natural é simplificar as derivadas e integrais necessárias no cálculo de diversas grandezas através de um sistema de coordenadas correspondente com a geometria do elemento. Este tipo de parametrização das aproximações envolve o mapeamento do elemento, definido no sistema de coordenadas naturais e normalmente denominado elemento padrão, para sua forma distorcida no sistema de coordenadas cartesianas. Este processo é completamente geral, requerendo somente que se estabeleça uma correspondência biunívoca entre os sistemas de coordenadas cartesianas $\langle X_1, X_2 \rangle$ e naturais $\langle \xi, \eta \rangle$, traduzida na relação:

$$\begin{Bmatrix} X_1 \\ X_2 \end{Bmatrix} = \begin{Bmatrix} X_1(\xi, \eta) \\ X_2(\xi, \eta) \end{Bmatrix} \quad (3.1)$$

A Figura 3. 2 ilustra este processo de parametrização.

Figura 3. 2 - Mapeamento unidimensional e bidimensional. Fonte: Almeida(2005).



Um elemento finito pode ser classificado em isoparamétrico, superparamétrico e subparamétrico de acordo com a ordem das funções de interpolação que aproximam a geometria da estrutura e a variável de interesse. Se as funções de interpolação da geometria são de ordem inferior às funções de interpolação da variável em estudo, o elemento é considerado subparamétrico. Caso as funções de interpolação da geometria sejam de ordem superior às utilizadas na aproximação da grandeza de interesse, tem-se o elemento superparamétrico.

3.3 Tipos de Elementos Finitos

De acordo com Zienkiewicz(1983) os sistemas de coordenadas naturais normalmente utilizados para parametrizar aproximações devem seguir a geometria do problema em análise. Podem, também, ser de aproximação linear ou de grau superior, a depender do grau das funções utilizadas para interpolar os campos sobre cada elemento. Em princípio, qualquer mapeamento ou parametrização pode ser utilizado para descrever a geometria de elementos. O conjunto de elementos é responsável por descrever a geometria do problema. Esses elementos, quanto ao domínio, podem ser classificados

como:

- Elementos unidimensionais: ou uniaxiais, são segmentos de reta (lineares) ou segmentos de curva.
- Elementos bidimensionais: são faces planas ou polígonos, quando as funções aproximativas são lineares ou superfícies curvas quando o grau do polinômio aproximador é de ordem superior. Usualmente utilizam-se geometrias triangulares ou quadrilaterais de discretização.
- Elementos tridimensionais: são sólidos, em forma de paralelepípedo ou tetraedros, por exemplo.

3.4 Método dos Elementos Finitos

Boa parte dos fenômenos da engenharia são modelados através de equações diferenciais. Muitas destas equações são impossíveis de serem solucionadas através de métodos analíticos devido as características inerentes a cada tipo de problema. Dentre os métodos numéricos para solução aproximada de equações diferenciais parciais destaca-se o Método dos Elementos Finitos (MEF). Este método considera a região de solução do problema formada por pequenos elementos, caracterizados por nós interligados entre si através de funções aproximativas. De acordo com Sánchez (2001) o domínio pode ser modelado por tais elementos em diferentes configurações, e permitem assim, a modelagem de formas geométricas das mais complexas. Por isso, pode ser aplicado em uma grande faixa de problemas de engenharia que envolve valores de contorno, e permite, por exemplo, o estudo dos deslocamentos e tensões em peças mecânicas, barragens, minas e estruturas das mais diversas formas. O método é também utilizado para determinar percolação, adensamento, pressão neutra, fluxo de calor, e muitas outras análises.

No âmbito da Engenharia de Estruturas o MEF é aplicado na determinação dos campos de deslocamentos, tensões e deformações de um sólido de geometria arbitrária sujeito a ações externas. A técnica permite análise do comportamento de uma estrutura cuja geometria, materiais e ações são, a priori, conhecidos. Segundo Azevedo (2003) a formulação pode ser baseada no método dos deslocamentos, em modelos de equilíbrio, ou em métodos híbridos e mistos. Sendo mais versátil e simples a versão em

deslocamentos.

Nesta formulação a resposta de cada elemento é determinada em termos de um número finito de graus de liberdade caracterizados como um valor de uma função, em um conjunto de pontos. Esses graus de liberdade são o número de movimentos possíveis independentes presentes em cada nó, tais como rotação e translação. A resposta do modelo matemático, finalmente, é obtida reunindo-se todos os elementos, Fellipa (2000). Isto é, no modelo de deslocamentos do MEF o domínio do problema é subdividido em elementos finitos onde o campo de deslocamentos é aproximado em cada elemento em função dos deslocamentos nodais. Assim é possível montar um sistema de equações algébricas que, quando resolvido, permite solucionar o problema.

A construção do modelo que melhor representa de modo discreto o domínio do problema pelo método é realizada seguindo, resumidamente, os seguintes passos:

- Pré-processamento: divisão do domínio do problema em elementos finitos. Estes elementos estão conectados pelos nós comuns e juntos aproximam a forma do domínio;
- Formulação dos elementos: desenvolvimento de equações que representam a influência de cada nó da discretização. Tais equações implicam em aproximar a grandeza de interesse em cada elemento por um polinômio que garanta condições de continuidade e completude entre os nós de cada elemento. Por questões de simplificação, essas funções podem ser mapeadas em outros sistemas de coordenadas;
- Montagem: obtenção do sistema global de equações a partir das contribuições de todos os elementos;
- Resolução do Sistema de equações, que representa algebricamente a condição de equilíbrio da estrutura;
- Pós-processamento: determinação de valores de interesse, tais como tensões e deformações, com base nos valores obtidos das variáveis básicas, deslocamento e força de superfície, além da visualização dos resultados.

3.5 Formulação em Deslocamentos do MEF aplicado à Elasticidade Plana

A formulação isoparamétrica em deslocamentos do Método dos Elementos Finitos (em versões linear e não linear geométrica) é brevemente descrita a seguir, e aplicada na determinação de soluções aproximadas do problema de elasticidade plana, definida pelos grupos de equações de equilíbrio, cinemáticas e constitutivas.

Mapeada a geometria do problema através da escolha de um elemento finito, determinados os materiais e ações presentes na estrutura aplica-se a formulação de acordo com a análise desejada.

3.5.1 Formulação bidimensional linear do Método dos Elementos Finitos

Na formulação linear do MEF adotam-se hipóteses simplificadoras para a idealização do comportamento da estrutura. Admite-se que o material é elástico linear, ou seja, as tensões relacionam-se linearmente com as deformações de modo que o corpo retoma sua forma original quando as tensões atuantes são retiradas. Continua-se também a admitir que sejam desprezíveis as forças de inércia e de amortecimento que se possam desenvolver durante o carregamento da estrutura (comportamento quase estático). E que as análises são feitas na configuração indeformada, em regime de pequenas deformações e deslocamentos (linearidade geométrica).

Sabendo que p representa os graus de liberdade para cada nó, nn o número de nós do elemento e i o respectivo nó, assume-se que o campo de deslocamentos de um elemento e pode ser descrito como o produto das funções de aproximação N_p pelos deslocamentos dos nós do elemento.

$$\mathbf{u}^e \approx \mathbf{N}_p \cdot \mathbf{d}^e \quad (3.1)$$

$$u_i^e(\xi, \eta) = \sum_{p=1}^{nn} N_p(\xi, \eta) \cdot d_{i(p)}^e \quad (3.2)$$

A matriz \mathbf{N} que reúne as funções de aproximação em coordenadas naturais é expressa através da equação (3.3) como:

$$\begin{bmatrix} N(\xi) \\ N(\eta) \end{bmatrix} = \begin{Bmatrix} N_1(\xi, \eta) & 0 & N_2(\xi, \eta) & 0 & \cdots & \cdots & N_{nn}(\xi, \eta) & 0 \\ 0 & N_1(\xi, \eta) & 0 & N_2(\xi, \eta) & \cdots & \cdots & 0 & N_{nn}(\xi, \eta) \end{Bmatrix} \quad (3.3)$$

E os deslocamentos nodais \mathbf{d}^e para cada elemento e , com 2 graus de liberdade de deslocamento nas direções 1 e 2 podem ser escritos como:

$$\mathbf{d}^e = \begin{Bmatrix} d_{11} & d_{12} \\ \vdots & \vdots \\ d_{(nn)1} & d_{(nn)2} \end{Bmatrix} \quad (3.4)$$

A condição de compatibilidade é imposta localmente para definir a aproximação do campo de deformações em cada elemento e permite expressar as deformações em termos das funções de interpolação e do deslocamento nodal. Aplicando o gradiente simétrico \mathbf{L} às funções de forma, matricialmente, obtém-se:

$$\boldsymbol{\varepsilon} \approx \mathbf{L}\mathbf{u}^e = \mathbf{L}\mathbf{N}\mathbf{d}^e \quad (3.5)$$

em que \mathbf{LN} origina a matriz \mathbf{B} que compreende as funções aproximativas dos elementos adotados, designada matriz de deformação-deslocamento, ou seja:

$$\mathbf{B} = \mathbf{LN} \quad (3.6)$$

Então \mathbf{B} pode ser reescrita como:

$$\mathbf{B} = \begin{Bmatrix} \frac{\partial N_1}{\partial X_1} & 0 & \frac{\partial N_2}{\partial X_1} & 0 & \cdots & \cdots & \frac{\partial N_p}{\partial X_1} & 0 \\ 0 & \frac{\partial N_1}{\partial X_2} & 0 & \frac{\partial N_2}{\partial X_2} & \cdots & \cdots & 0 & \frac{\partial N_p}{\partial X_2} \\ \frac{\partial N_1}{\partial X_2} & \frac{\partial N_1}{\partial X_1} & \frac{\partial N_2}{\partial X_2} & \frac{\partial N_2}{\partial X_1} & \cdots & \cdots & \frac{\partial N_{nn}}{\partial X_2} & \frac{\partial N_p}{\partial X_1} \end{Bmatrix} \quad (3.7)$$

Como as funções de interpolação de \mathbf{N} foram escritas em coordenadas paramétricas é necessário o mapeamento com as coordenadas cartesianas. O conjunto de equações responsáveis por essa transformação e designado por matriz de jacobiana \mathbf{J} .

Pela regra da cadeia obtém-se:

$$\frac{\partial N_p}{\partial \xi} = \frac{\partial N_p}{\partial X_1} \frac{\partial X_1}{\partial \xi} + \frac{\partial N_p}{\partial X_2} \frac{\partial X_2}{\partial \xi}$$

$$\frac{\partial N_p}{\partial \eta} = \frac{\partial N_p}{\partial X_1} \frac{\partial X_1}{\partial \eta} + \frac{\partial N_p}{\partial X_2} \frac{\partial X_2}{\partial \eta}$$
(3.8)

Portanto,

$$\frac{\partial X_i}{\partial \xi} = \sum_{p=1}^{nn} \frac{\partial N_p}{\partial \xi} \cdot X_{i(p)}$$

$$\frac{\partial X_i}{\partial \eta} = \sum_{p=1}^{nn} \frac{\partial N_p}{\partial \eta} \cdot X_{i(p)}$$
(3.9)

Então o *Jacobiano* é encontrado por:

$$J = \begin{pmatrix} \frac{\partial N_1}{\partial \xi} & \frac{\partial N_2}{\partial \xi} & \frac{\partial N_3}{\partial \xi} & \dots & \frac{\partial N_{nn-1}}{\partial \xi} & \frac{\partial N_{nn}}{\partial \xi} \\ \frac{\partial N_1}{\partial \eta} & \frac{\partial N_2}{\partial \eta} & \frac{\partial N_3}{\partial \eta} & \dots & \frac{\partial N_{nn-1}}{\partial \eta} & \frac{\partial N_{nn}}{\partial \eta} \end{pmatrix} \cdot \begin{pmatrix} X_{1(1)} & X_{2(1)} \\ \vdots & \vdots \\ X_{1(nn)} & X_{2(nn)} \end{pmatrix}$$
(3.10)

A equação que governa o elemento é estabelecida interpretando-a como uma condição de equilíbrio de forças nodais. Assim os deslocamentos nodais e globais devem ser encontrados de maneira a satisfazer a formulação fraca para a elasticidade linear. Através da manipulação da formulação é possível extrair a matriz de rigidez \mathbf{K} que descreve o comportamento do material e o vetor \mathbf{F} de ações sobre o corpo, como segue, respectivamente:

$$\mathbf{K} = \int_{\Omega} \mathbf{B}_L^T \cdot \mathbf{D} \cdot \mathbf{B}_L d\Omega$$
(3.11)

onde \mathbf{D} é a matriz que caracteriza as propriedades mecânicas do material, designada matriz constitutiva.

$$\mathbf{F} = \mathbf{F}_b + \mathbf{F}_t + \mathbf{F}_n$$
(3.12)

O vetor de ações resultantes \mathbf{F} é dado como a soma de todas as forças que agem sobre o

corpo, em que as forças nodais equivalentes às forças de corpo são:

$$\mathbf{F}_b = \int_{\Omega} \mathbf{N}^T \mathbf{b} d\Omega \quad (3.13)$$

Já as forças nodais equivalentes às forças de superfície, essas são obtidas por:

$$\mathbf{F}_t = \int_{\Gamma} \mathbf{N}^T \mathbf{t} d\Gamma \quad (3.14)$$

Representam-se as forças concentradas aplicadas nos nós do elemento por meio de F_n .

Para elementos planos \mathbf{K} a matriz de rigidez pode ser reescrita em termos de coordenadas paramétricas como segue:

$$\begin{aligned} \mathbf{K} &= \int_{X_2} \int_{X_1} \mathbf{B}_L^T \cdot \mathbf{D} \cdot \mathbf{B}_L dX_1 dX_2 = \int_{-1}^1 \int_{-1}^1 \mathbf{B}_L^T \mathbf{D} \cdot \mathbf{B}_L |J| \cdot d\xi d\eta = \\ &\approx \sum_{i=1}^{npg} \sum_{j=1}^{npg} \mathbf{B}_L^T \cdot \mathbf{D} \cdot \mathbf{B}_L \cdot |J| \cdot w_i \cdot w_j \end{aligned} \quad (3.15)$$

em que w representa o vetor de valores nodais das funções peso, impostas durante a formulação de acordo com o número de pontos de Gauss npg necessários à modelagem o elemento.

De acordo com Azevedo (2003) a solução aproximada em cada elemento está sujeita à condição de ser cinematicamente admissível de modo a satisfazer localmente à condição de fronteira cinemática da malha de elementos finitos e, ainda, à condição equivalente entre elementos. Satisfeitas essas condições é possível expressar os deslocamentos e ações globais por meio da relação diretamente proporcional entre \mathbf{K} e \mathbf{F} .

3.5.2 Formulação bidimensional não linear geométrica do Método dos Elementos Finitos

Apresenta-se a formulação do método aplicado a problemas geometricamente não lineares, em regime de deformação finita, conforme mostrado em Cavalcante (2013). Sabendo que p representa os graus de liberdade para cada nó, nn o número de nós do elemento e i o respectivo nó, assume-se que o campo de deslocamentos pode ser descrito

como o produto de funções de aproximação N_p pelos deslocamentos nodais. Sejam dois instantes de tempo subsequentes, separados pelo incremento Δt . A variação de deslocamento neste intervalo de tempo pode ser definida como segue:

$$u_i^t(\xi, \eta) = \sum_{p=1}^{nn} N_p(\xi, \eta) \cdot d_{i(p)}^t \quad (3.16)$$

$$u_i^{t+\Delta t}(\xi, \eta) = \sum_{p=1}^{nn} N_p(\xi, \eta) \cdot d_{i(p)}^{t+\Delta t} \quad (3.17)$$

$$\Delta u_i(\xi, \eta) = \sum_{p=1}^{nn} N_p(\xi, \eta) \cdot \Delta d_{i(p)} \quad (3.18)$$

O vetor que contém os deslocamentos nodais é expresso como:

$$\Delta d_{i(p)} = d_{i(p)}^{t+\Delta t} - d_{i(p)}^t \quad (3.19)$$

Reescrevendo as equações acima na forma matricial:

$$\begin{aligned} \mathbf{u}^t(\xi, \eta) &= \mathbf{N}(\xi, \eta) \cdot \mathbf{d}^t \\ \mathbf{u}^{t+\Delta t}(\xi, \eta) &= \mathbf{N}(\xi, \eta) \cdot \mathbf{d}^{t+\Delta t} \\ \Delta \mathbf{u}(\xi, \eta) &= \mathbf{N}(\xi, \eta) \cdot \Delta \mathbf{d} \end{aligned} \quad (3.20)$$

Sabendo que \mathbf{N} é a matriz que reúne as funções de aproximação

$$\begin{aligned} &\begin{bmatrix} N(\xi) \\ N(\eta) \end{bmatrix} \\ &= \left\{ \begin{array}{cccccc} N_1(\xi, \eta) & 0 & N_2(\xi, \eta) & 0 & \dots & \dots & N_p(\xi, \eta) & 0 \\ 0 & N_1(\xi, \eta) & 0 & N_2(\xi, \eta) & \dots & \dots & 0 & N_p(\xi, \eta) \end{array} \right\} \end{aligned} \quad (3.21)$$

onde em cada nó o deslocamento é $\mathbf{d}^t = \{d_1^t \ d_2^t \ d_3^t \ \dots \ d_{nn}^t\}^T$

Tomando que há duas coordenadas representando deslocamento nas direções X_1 e X_2 apenas, obtém-se a matriz de deslocamentos:

$$\mathbf{d}_{(p)}^t = \begin{Bmatrix} d_{11}^t & d_{12}^t \\ \vdots & \vdots \\ d_{(nn)1}^t & d_{(nn)2}^t \end{Bmatrix} \quad (3.22)$$

De maneira análoga

$$\mathbf{d}^{t+\Delta t} = \{\mathbf{d}_1^{t+\Delta t} \mathbf{d}_2^{t+\Delta t} \mathbf{d}_3^{t+\Delta t} \dots \mathbf{d}_{nn}^{t+\Delta t}\}^T \therefore \mathbf{d}_{(p)}^{t+\Delta t} = \begin{Bmatrix} d_{11}^{t+\Delta t} & d_{12}^{t+\Delta t} \\ \vdots & \vdots \\ d_{(nn)1}^{t+\Delta t} & d_{(nn)2}^{t+\Delta t} \end{Bmatrix} \quad (3.23)$$

Para a variação no deslocamento em cada nó,

$$\Delta \mathbf{d} = \{\Delta \mathbf{d}_1 \Delta \mathbf{d}_2 \Delta \mathbf{d}_3 \dots \Delta \mathbf{d}_{nn}\}^T \therefore \Delta \mathbf{d}_{(p)} = \begin{Bmatrix} \Delta d_1 & \Delta d_1 \\ \vdots & \vdots \\ \Delta d_{nn} & \Delta d_{nn} \end{Bmatrix} \quad (3.24)$$

Análogo à (3.5) e (3.6), as deformações podem ser expressas em termos das funções de aproximação e deslocamentos nodais. Desta forma para uma análise não linear geométrica deve-se considerar a matriz deformação-deslocamento composta por um termo \mathbf{B}_L e outro \mathbf{B}_{NL} como define:

$$\mathbf{B}_L = \mathbf{B}_{L0} + \mathbf{B}_{L1} \quad (3.25)$$

$$\mathbf{B}_{L0} = \begin{Bmatrix} \frac{\partial N_1}{\partial X_1} & 0 & \frac{\partial N_2}{\partial X_1} & 0 & \dots & \dots & \frac{\partial N_p}{\partial X_1} & 0 \\ 0 & \frac{\partial N_1}{\partial X_2} & 0 & \frac{\partial N_2}{\partial X_2} & \dots & \dots & 0 & \frac{\partial N_p}{\partial X_2} \\ \frac{\partial N_1}{\partial X_2} & \frac{\partial N_1}{\partial X_1} & \frac{\partial N_2}{\partial X_2} & \frac{\partial N_2}{\partial X_1} & \dots & \dots & \frac{\partial N_{nn}}{\partial X_2} & \frac{\partial N_p}{\partial X_1} \end{Bmatrix} \quad (3.26)$$

Consideremos o tensor gradiente de deslocamento, desprezível em análises em pequenas deformações e deslocamentos, expresso como:

$$H_{ij}^t = \sum_{p=1}^{nn} \frac{\partial N_p}{\partial X_j} \cdot d_{i(p)}^t \quad (3.27)$$

Então,

$B_{NL} =$

$$\left\{ \begin{array}{cccc} H_{11}^t \frac{\partial N_1}{\partial X_1} & H_{21}^t \frac{\partial N_1}{\partial X_1} & \dots & H_{11}^t \frac{\partial N_{nn}}{\partial X_1} & H_{21}^t \frac{\partial N_{nn}}{\partial X_1} \\ H_{12}^t \frac{\partial N_1}{\partial X_2} & H_{22}^t \frac{\partial N_1}{\partial X_2} & \dots & H_{12}^t \frac{\partial N_{nn}}{\partial X_2} & H_{22}^t \frac{\partial N_{nn}}{\partial X_2} \\ H_{11}^t \frac{\partial N_2}{\partial X_2} + H_{12}^t \frac{\partial N_2}{\partial X_1} & H_{21}^t \frac{\partial N_2}{\partial X_2} + H_{22}^t \frac{\partial N_2}{\partial X_1} & \dots & H_{11}^t \frac{\partial N_{nn}}{\partial X_2} + H_{12}^t \frac{\partial N_{nn}}{\partial X_1} & H_{21}^t \frac{\partial N_{nn}}{\partial X_2} + H_{22}^t \frac{\partial N_{nn}}{\partial X_1} \end{array} \right\} \quad (3.28)$$

O componente não linear da matriz deformação-deslocamento é dado por:

$$B_{NL} = \left\{ \begin{array}{ccccccccc} \frac{\partial N_1}{\partial X_1} & \frac{\partial N_1}{\partial X_2} & 0 & 0 & \frac{\partial N_2}{\partial X_1} & \frac{\partial N_2}{\partial X_2} & 0 & 0 & \dots & \frac{\partial N_{nn}}{\partial X_1} & \frac{\partial N_{nn}}{\partial X_2} & 0 & 0 \\ 0 & 0 & \frac{\partial N_1}{\partial X_1} & \frac{\partial N_1}{\partial X_2} & 0 & 0 & \frac{\partial N_2}{\partial X_1} & \frac{\partial N_2}{\partial X_2} & \dots & 0 & 0 & \frac{\partial N_{nn}}{\partial X_1} & \frac{\partial N_{nn}}{\partial X_2} \end{array} \right\} \quad (3.29)$$

Dada a formulação paramétrica, existe um mapeamento entre as coordenadas cartesianas e as coordenadas paramétricas, dado pela matriz de transformação jacobiana J .

Pela regra da cadeia obtém-se:

$$\begin{aligned} \frac{\partial N_p}{\partial \xi} &= \frac{\partial N_p}{\partial X_1} \frac{\partial X_1}{\partial \xi} + \frac{\partial N_p}{\partial X_2} \frac{\partial X_2}{\partial \xi} \\ \frac{\partial N_p}{\partial \eta} &= \frac{\partial N_p}{\partial X_1} \frac{\partial X_1}{\partial \eta} + \frac{\partial N_p}{\partial X_2} \frac{\partial X_2}{\partial \eta} \end{aligned} \quad (3.30)$$

Logo,

$$\begin{pmatrix} \frac{\partial N_p}{\partial \xi} \\ \frac{\partial N_p}{\partial \eta} \end{pmatrix} = \begin{bmatrix} \frac{\partial X_1}{\partial \xi} & \frac{\partial X_2}{\partial \xi} \\ \frac{\partial X_1}{\partial \eta} & \frac{\partial X_2}{\partial \eta} \end{bmatrix} \cdot \begin{pmatrix} \frac{\partial N_p}{\partial X_1} \\ \frac{\partial N_p}{\partial X_2} \end{pmatrix} \therefore \begin{pmatrix} \frac{\partial N_p}{\partial X_1} \\ \frac{\partial N_p}{\partial X_2} \end{pmatrix} = J^{-1} \cdot \begin{pmatrix} \frac{\partial N_p}{\partial \xi} \\ \frac{\partial N_p}{\partial \eta} \end{pmatrix} \quad (3.31)$$

Portanto

$$\frac{\partial X_i}{\partial \xi} = \sum_{p=1}^{nn} \frac{\partial N_p}{\partial \xi} \cdot X_{i(p)} \quad (3.32)$$

$$\frac{\partial X_i}{\partial \eta} = \sum_{p=1}^{nn} \frac{\partial N_p}{\partial \eta} \cdot X_{i(p)}$$

Então o Jacobiano é dado como

$$J = \begin{pmatrix} \frac{\partial N_1}{\partial \xi} & \frac{\partial N_2}{\partial \xi} & \frac{\partial N_3}{\partial \xi} & \dots & \frac{\partial N_{nn-1}}{\partial \xi} & \frac{\partial N_{nn}}{\partial \xi} \\ \frac{\partial N_1}{\partial \eta} & \frac{\partial N_2}{\partial \eta} & \frac{\partial N_3}{\partial \eta} & \dots & \frac{\partial N_{nn-1}}{\partial \eta} & \frac{\partial N_{nn}}{\partial \eta} \end{pmatrix} \cdot \begin{pmatrix} X_{1(1)} & X_{2(1)} \\ \vdots & \vdots \\ X_{1(nn)} & X_{2(nn)} \end{pmatrix} \quad (3.33)$$

Inicialmente as coordenadas \mathbf{X} equivalem a \mathbf{x} na configuração deformada, ou seja, as coordenadas nos nós. Assim equação que governa o elemento é estabelecida interpretando-a como uma condição de equilíbrio de forças nodais equivalentes.

De acordo com Cavalcante(2013) a matriz de rigidez que descreve o comportamento do material é dada pela soma das componentes de rigidez elástica (linear) e geométrica (não linear):

$$\mathbf{K} = \mathbf{K}_L + \mathbf{K}_{NL} \quad (3.34)$$

A matriz geométrica que descreve o termo não linear da matriz de rigidez é dado por:

$$\begin{aligned} \mathbf{K}_L &= \int_{X_2} \int_{X_1} \mathbf{B}_L^T \cdot \hat{\mathbf{D}}^t \cdot \mathbf{B}_L dX_1 dX_2 = \int_{-1}^1 \int_{-1}^1 \mathbf{B}_L^T \cdot \hat{\mathbf{D}}^t \cdot \mathbf{B}_L |J| \cdot d\xi d\eta = \\ &\approx \sum_{i=1}^{npg} \sum_{j=1}^{npg} \mathbf{B}_L^T \cdot \hat{\mathbf{D}}^t \cdot \mathbf{B}_L \cdot |J| \cdot w_i \cdot w_j \end{aligned} \quad (3.35)$$

O componente não linear é descrito como:

$$\begin{aligned} \mathbf{K}_{NL} &= \int_{X_2} \int_{X_1} \mathbf{B}_{NL}^T \cdot \mathbf{T}^t \cdot \mathbf{B}_{NL} dX_1 dX_2 = \int_{-1}^1 \int_{-1}^1 \mathbf{B}_{NL}^T \cdot \hat{\mathbf{T}}^t \cdot \mathbf{B}_{NL} |J| \cdot d\xi d\eta = \\ &\approx \sum_{i=1}^{npg} \sum_{j=1}^{npg} \mathbf{B}_{NL}^T \cdot \hat{\mathbf{T}}^t \cdot \mathbf{B}_{NL} \cdot |J| \cdot w_i \cdot w_j \end{aligned} \quad (3.36)$$

onde \mathbf{D} caracteriza as propriedades mecânicas do material, e \mathbf{w} a matriz de pesos.

Os valores de \mathbf{w} estão relacionados à quantidade de pontos de Gauss np_g utilizados para integração.

A matriz \mathbf{T} é o segundo tensor de Piola-Kirchhoff. Este tensor é obtido a partir do primeiro tensor de Piola-Kirchhoff, uma representação quadrada e simétrica advinda do tensor de Cauchy.

O vetor de forças resultantes é dado como a soma de todas as forças que agem sobre o corpo, de modo a atender a condição de equilíbrio, ou seja:

$$\begin{aligned} \mathbf{F}_{int}^t &= \int_{X_2} \int_{X_1} \mathbf{B}_{NL}^T \cdot \mathbf{T}^t dX_1 dX_2 = \int_{-1}^1 \int_{-1}^1 \mathbf{B}_{NL}^T \cdot \mathbf{T}^t \cdot |\mathbf{J}| \cdot d\xi d\eta = \\ &\approx \sum_{i=1}^{np_g} \sum_{j=1}^{np_g} \mathbf{B}_{NL}^T \cdot \mathbf{T}^t \cdot |\mathbf{J}| \cdot w_i \cdot w_j \end{aligned} \quad (3.37)$$

Uma vez que o carregamento será admitido independente da configuração deformada do sólido, o vetor de forças externas $\mathbf{F}_{ext}^{t+\Delta t}$ pode ser calculado usando um procedimento semelhante aquele adotado em análises. No regime de pequenos deslocamentos, baseado no conceito de forças nodais estatisticamente equivalentes.

No caso de uma análise incremental, o carregamento externo deve ser aplicado aos poucos. Logo, o seguinte procedimento pode ser adotado utilizando o parâmetro de carregamento $\lambda^{t+\Delta t}$.

$$\mathbf{F}_{ext}^{t+\Delta t} = \lambda^{t+\Delta t} \cdot \mathbf{F}_{ext} \quad (3.38)$$

\mathbf{F}_{ext} é o vetor de forças externas calculado utilizando o carregamento final (total) aplicado à estrutura e $0 < \lambda^{t+\Delta t} \leq 1$ (Parâmetro de carregamento).

No caso de uma taxa de aplicação uniforme do carregamento, tem-se:

$$\lambda^{t+\Delta t} = \lambda^t + \Delta\lambda \text{ onde } \Delta\lambda = 1/(\text{número de incrementos}) \quad (3.39)$$

A solução aproximada em cada elemento deve ser construída de modo a satisfazer localmente à condição de fronteira cinemática da malha de elementos finitos e, ainda, à condição equivalente entre elementos.

4 REDES NEURAIS ARTIFICIAIS

4.1 Conceitos

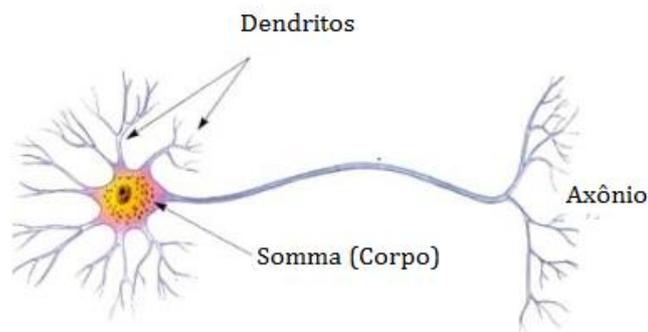
A ciência descreve o cérebro humano como o processador ideal devido à alta capacidade de aprender, generalizar, memorizar e criar, além de processar informações incompletas, imperfeitas e com ruído. É o maior órgão do sistema nervoso e possui de 10 a 100 bilhões de células nervosas que funcionam de modo paralelo e distribuído. Atualmente, programas de computador podem emular a maneira como o cérebro resolve problemas, como uma rede interconectada de neurônios que trabalham paralelamente. Lastiti *et al.* (2004) define as Redes Neurais Artificiais como técnicas computacionais que apresentam um modelo matemático inspirado na estrutura neural de organismos inteligentes, onde o processo de aprendizagem ocorre através da experiência. Convencionou-se assim, chamar de Redes Neurais Artificiais (RNA) toda estrutura de processamento, *software* ou *hardware*, que utiliza o conexionismo como paradigma.

4.2 Neurônio Biológico

O neurônio biológico é uma célula nervosa, principal elemento do sistema nervoso, Figura 4.1, e quando interconectados entre si os neurônios formam os circuitos neurais. Estes circuitos constituem o sistema neural, responsável por detectar estímulos externos e internos, que podem ser físicos ou químicos e desencadear respostas musculares e glandulares. Segundo Tafner (1998), existem vários tipos de neurônios, sendo formados pelos seguintes componentes:

- Dendritos: recebem estímulos;
- Corpo (soma): coleta e combina informações (Na^+ e K^+); e
- Axônio: fibra que transmite os estímulos.

Figura 4. 1 - Neurônio humano. Fonte: adaptado Braga et. al (1999).

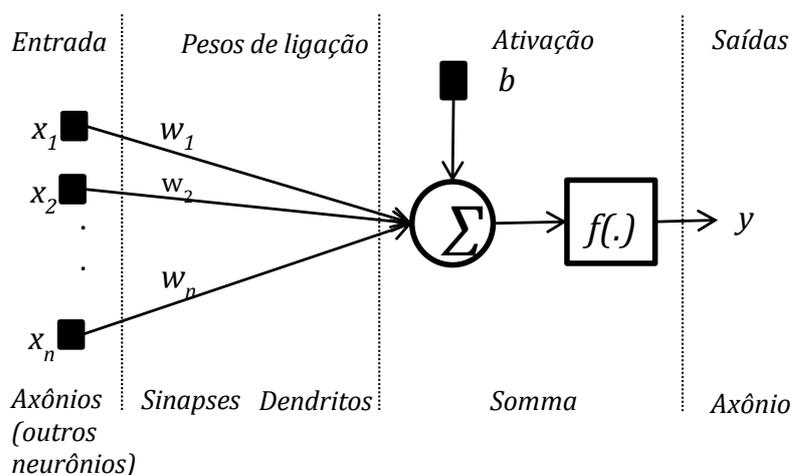


O sinal do neurônio é transmitido à outro através de reação eletroquímica entre as conexões (sinapses) existentes entre essas células. Substâncias químicas transmissoras são lançadas dessas sinapses e entram pelos dendritos e provocam alteração do potencial elétrico do corpo da célula. Quando o potencial chega a um limiar, um pulso elétrico é transmitido pelo axônio às outras sinapses espalhando o pulso longo das conexões existentes, Medeiros (2003).

4.3 Neurônio Artificial

O neurônio artificial surgiu como uma tentativa de descrever a complexa estrutura de processamento do neurônio humano. A estrutura básica do neurônio artificial é representada por meio de analogia deste com o neurônio biológico, conforme Figura 4. 2.

Figura 4. 2 - Neurônio artificial. Fonte: elaborada pela autora.



Os neurônios artificiais são simples unidades de processamento que podem

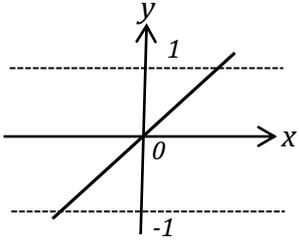
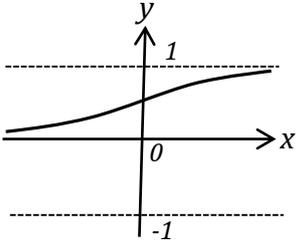
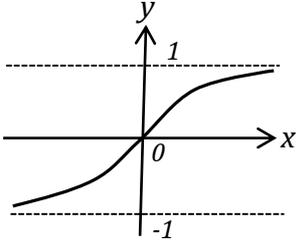
receber várias entradas de outros neurônios, através dos dendritos, produzindo um único sinal de saída que se propaga pelo axônio para o neurônio seguinte, Martins (1997). A descrição de um neurônio artificial pode ser feita por uma equação matemática (4.1) muito simples que mapeia o valor de saída em função dos valores de entrada na forma:

$$y_i = \sum_{i=1}^{NN} w_i x_i \quad (4.1)$$

Associado a cada entrada x_i , tem-se um valor w_i , designado por peso de ligação. À soma ponderada das entradas dá-se o nome de ativação do neurônio, que depois de transformada por uma função de ativação ou de transferência, $f(\cdot)$, possibilita a obtenção da saída y . O *bias*, b , permite obter uma saída não nula quando todas as entradas x_i forem iguais a zero.

Existem vários tipos de função de ativação as quais produzem diferentes saídas e deve ser escolhida de acordo com o tipo de aplicação. Tais funções determinam o nível de transformação de um neurônio perante as suas entradas, Roseiro (2004). Dentre as quais se destacam: linear, logística e tangente hiperbólica conforme Tabela 4. 1.

Tabela 4. 1 – Funções de ativação usuais.

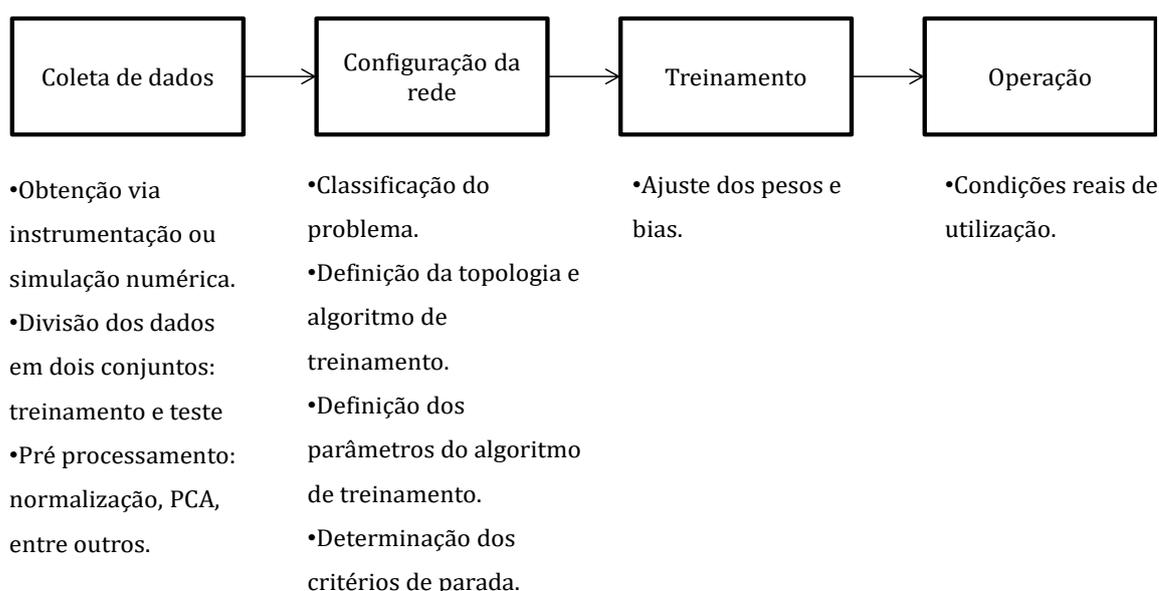
Linear	Logística	Tangente Hiperbólica
$y = a \cdot x$	$y = \frac{1}{1 + e^{-\gamma \cdot x}}$	$y = \frac{1 - e^{-\gamma \cdot x}}{1 + e^{-\gamma \cdot x}}$
$[-\infty, \infty]$	$[0, 1]$	$[-1, 1]$
		

Para efeitos de eficiência da Rede Neural Artificial a função de ativação deve ser contínua, monotônica, não linear e diferenciável em qualquer ponto. Essas condições

garantem a limitação superior ou inferior dos dados de entrada para que os valores calculados à saída dos neurônios não apresentem comportamentos instáveis. Os pesos de ligação são parâmetros ajustáveis da RNA que representam a ponderação de um neurônio a outro ao longo da transferência da informação da entrada à saída. As entradas de uma determinada camada são formadas pela multiplicação do peso associado a uma determinada ligação e a saída do neurônio da camada anterior correspondente a tal ligação, exceto para camada de entrada. Os valores dos pesos após o ajuste representam a memória do fenômeno e caracterizam o processo de aprendizagem da RNA, assim como representam o grau de interatividade entre dois neurônios artificiais.

O uso do neurônio artificial pode ocorrer em duas fases sequenciais e distintas. A primeira, o treinamento, consiste no ajuste dos pesos e *bias* de maneira a mapear os sinais de entrada e saída da rede. E a segunda fase, de simulação ou operação, que consiste na aplicação da rede no problema, onde a rede deverá se apresentar capaz de generalizar a solução e fornecer saída para dados não contidos no domínio de dados usados no treinamento. Entretanto, o processo de obtenção da solução não ocorre de maneira trivial. O projeto eficiente das redes envolve uma série de considerações e recomenda-se que sejam cumpridas etapas, conforme sugerido no diagrama da Figura 4. 3.

Figura 4. 3 – Etapas de desenvolvimento de aplicações usando RNA. Fonte: elaborada pela autora.



Além das considerações já mencionadas, o projetista deve atentar, durante desenvolvimento das aplicações, para que as RNA cubram amplamente o domínio do problema e atendam as exceções e condições limites. O ajuste dos pesos durante o treinamento deve ser interrompido apenas quando a rede apresentar considerável

capacidade de generalização de acordo com o critério de parada e a rede é considerada de boa convergência quando a saída apresenta erros abaixo da tolerância pré estabelecidas durante a modelagem do fenômeno.

4.4 Breve Histórico

De acordo com Braga *et al.* (1999) o primeiro modelo artificial de um neurônio biológico surgiu em 1943, em artigos de Warren McCulloch, neurologista e psiquiatra, e Walter Pitts, matemático. Os autores apresentaram um modelo matemático no qual definiram um neurônio artificial em analogia com um neurônio biológico.

Hebb (1949) introduziu o conceito de aprendizagem através do seu livro “*The Organization of Behavior*”. O princípio “quando a entrada sináptica e a saída de um neurônio estão ambas ativas, ocorre uma mudança física caracterizada pelo aumento na força dessa ligação”, embasou os algoritmos de treinamento e ficou conhecido por Regra de Hebb.

Rosenblatt (1958) criou o *Perceptron*, modelo baseado na analogia com a retina do olho. O *Perceptron* consiste numa saída descrita como a soma ponderada de entradas.

Widrow e Hoff (1960) apresentaram o modelo ADALINE (*ADaptive LINear Element*), conhecido como regra delta e também designado como *least mean square rule* (LMS). Este modelo é capaz de ordenar padrões em duas categorias através de regra de aprendizagem baseada no método dos mínimos quadrados.

Entretanto, Minsky e Papert (1969) demonstraram que as regras até então eram aplicáveis apenas a problemas ditos linearmente separáveis.

Dadas as limitações identificadas e inerentes às tecnologias da época não ocorreram progressos significativos nesta área até que Hopfield (1982), Prêmio Nobel da Física, retomou as discussões sobre o assunto e demonstrou que uma rede opera com princípio de minimização de erro associado a cada problema.

Rumelhart *et al.* (1986) propuseram então o algoritmo *backpropagation* para treinar redes múltiplas camadas. O *backpropagation* tornou-se um algoritmo popular e de fácil implementação o que contribuiu para solucionar o problema de classificação de problemas não linearmente separáveis e ampliou o uso de Redes Neurais.

Desde então as redes neurais têm evoluído continuamente com o propósito de generalizar a aplicação em várias áreas, como a biologia, investigação policial, medicina, economia e finanças, engenharia, entre outras.

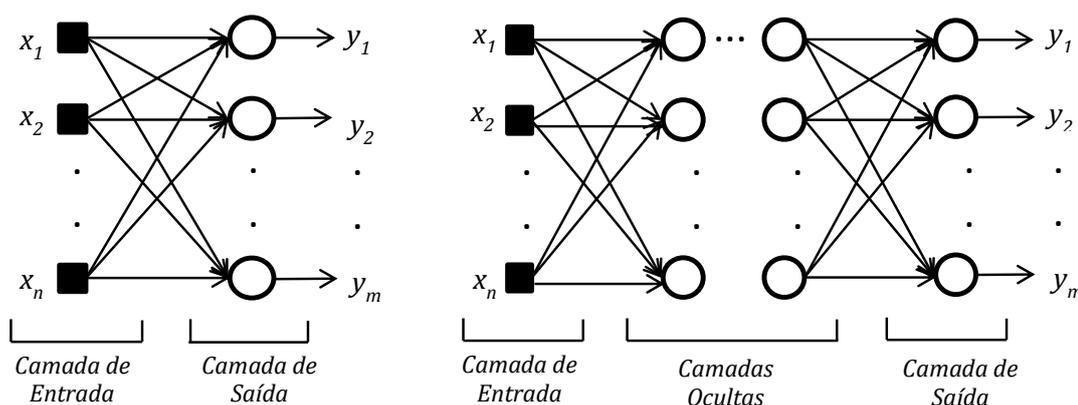
4.5 Arquitetura de Redes Neurais

A estrutura de uma Rede Neural Artificial determina as funções para as quais a rede melhor se aplica. As redes são classificadas de acordo com a sua topologia, ou seja, de acordo com o número de neurônios, as ligações entre neurônios e as funções de ativação empregadas.

Quanto ao número de camadas, Figura 4. 4, a rede pode ser classificada como:

- Redes de camada única: apresentam apenas neurônios de entrada e camada de neurônios saída, ou seja, não existem camadas intermediárias;
- Redes múltiplas camadas (MLP- *Multilayer Perceptron*): apresentam camadas intermediárias ou ocultas entre neurônios de entrada e camada de saída.

Figura 4. 4 – Representação de rede em camada única e múltiplas camadas, respectivamente. Fonte: elaborada pela autora.



Quanto à conexão entre os nós a rede pode ser classificada como:

- *Feedforward* ou acíclica: sinais projetados sempre para frente. A saída de um neurônio é a entrada para um neurônio localizado em camada posterior a da localização do neurônio;
- *Feedback*, cíclica ou recorrente: ocorre retorno dos sinais através de realimentação da saída nas entradas.

Quanto ao comportamento:

- Estáticas: a estrutura de camadas e neurônios definida não é alterada no decorrer do treinamento. Apenas os pesos sofrem alterações;
- Dinâmicas: neste tipo de redes ocorrem alterações em nível do número de neurônios e de ligações.

A arquitetura a ser empregada depende do tipo de problema a solucionar, seja um problema de classificação, de aproximação de funções, séries temporais, dentre outros.

4.6 Aprendizagem

O aprendizado caracteriza-se como a memória do fenômeno modelado. Ocorre através da aplicação de regras de associação entre variáveis, denominadas de algoritmo de aprendizagem. Os algoritmos de aprendizagem do projeto de Redes Neurais Artificiais são estruturados. Portanto, de acordo com a linha de aplicação e alguns detalhes da arquitetura da rede podem-se formar diferentes subáreas de aprendizagem. Em geral, podem-se identificar classes de arquitetura de funcionamento de acordo com o tipo de aprendizado, como sugere Lins (2002). A seguir têm-se os modelos mais utilizados: os supervisionados e os não supervisionados.

4.6.1 Aprendizado não supervisionado

Neste tipo de treinamento o ajuste dos pesos da rede é feito de modo a que a rede responda de modo semelhante quando sujeita a estímulos semelhantes, Barreto (2002).

Para o treinamento inexistem saídas, apenas entradas e pesos. Sempre que se apresenta uma nova entrada à Rede Neural Artificial, esta deve identificar as similaridades contidas nos conjuntos de amostras apresentados de modo a estabelecer uma harmonia com as regularidades entre entrada ao fornecer a resposta. Considera-se que a rede se encontra bem treinada quando apresenta capacidade de reconhecer todas as entradas introduzidas durante o treinamento. Esse tipo de aprendizado possui a necessidade de redundância na entrada, uma vez que, pode não reconhecer quaisquer padrões ou características nos dados, como é o caso das séries temporais. São exemplos, as redes competitivas, Kohonen e Hopfield.

4.6.2 Aprendizado supervisionado

É o treinamento mais utilizado nos diferentes tipos de Redes Neurais existentes. Neste método os pesos são continuamente ajustados a partir de regras baseadas na minimização do erro, nos quais os valores de saída da rede são comparados à valores esperados. Normalmente são focados no ajuste de valores de peso que permitam a classificação proposta no conjunto de treinamento constituído por pares de vetores entrada-saída. Este processo tem em vista encontrar o melhor conjunto de valores dos pesos que minimize o erro entre os valores conhecidos e os valores previstos pela rede.

Exemplos importantes deste tipo de treinamento são a regra de aprendizagem do *Perceptron*, a regra Delta e o algoritmo de retro propagação (*backpropagation*). Pela sua importância nesta dissertação, apresenta-se com mais detalhe este último algoritmo aplicado ao treinamento de redes multicamada de avanço simples (*Multilayer Feedforward Network*).

4.7 Redes de Camadas Múltiplas (MLP – *Multilayer Perceptron*)

Essa classe é composta de camada de entrada, camadas intermediárias ou ocultas e camada de saída. “A função dos neurônios ocultos é intervir entre a entrada externa e a saída de modo a tornar a rede capaz de extrair estatísticas de ordem elevada”, Haykin (1999). Na camada intermediária é onde ocorre a maior parte do processamento de informações.

Considera-se uma rede com M camadas, sendo a entrada da rede, a primeira camada $m = 1$, que apenas transporta a informação de entrada de cada neurônio para a camada seguinte, não tendo qualquer função de transformação, e a saída, a última camada $m = M$. As camadas restantes são designadas camadas intermediárias. Considera-se ainda que N_m representa o número de neurônios da camada m e que x_i^m representa a saída do neurônio i da camada m . Utilizando as definições (4.1), obtêm-se as seguintes expressões para o caso geral das camadas intermediárias e saída $m = 2, \dots, M$.

$$a_i^{(m)} = \sum_{j=1}^{N_{m-1}} w_{ij}^{(m-1)} x_j^{(m-1)} + b_i^{(m)} \quad (4.2)$$

4.8 Base Radial

Segundo Silva *et al.* (2007), “diferentemente das MLP, as quais podem ser compostas de várias camadas intermediárias, a estrutura típica de uma rede de funções de base radial (RBF) é composta por apenas uma, onde as funções de ativação são do tipo gaussiana”.

4.9 Algoritmo de Retro Propagação (*Backpropagation*)

O mais popular dos algoritmos ficou conhecido como *backpropagation*. Trata-se da extensão da regra delta Haykin (1999). Tornou-se o algoritmo mais utilizado para treinamento de RNA por atender a demanda de ajuste da camada intermediária e ser de fácil implementação .

Durante o treinamento da rede o algoritmo funciona em duas fases: em uma primeira fase os padrões de entrada são propagados através da rede, obtendo-se a resposta produzida pela camada de saída. Em uma segunda fase, utiliza-se uma propagação do erro (diferença entre as saídas desejada e produzida pela rede) da camada de saída para as camadas intermediárias de forma atualizar os pesos das conexões.

Para uma rede neural multicamadas com M camadas e N_m neurônios, o erro da resposta do neurônio i da camada de saída para o padrão de treino k , é dado pela diferença entre a saída produzida pela rede $x_i^{(M)}$ e a saída desejada x_i^d .

$$e_i^k = (x_i^d - x_i^{(M)})_k = (x_i^d - x_i)_k \quad (4.4)$$

Este algoritmo baseia-se na redução do erro quadrático entre as saídas desejada e obtida. Portanto, o erro para o k -ésimo conjunto de dados de entrada e saída do conjunto de treinamento:

$$E_k = \sum_{i=1}^{N_m} \left[\frac{1}{2} (e_i^k)^2 \right]_k \quad (4.5)$$

Para P conjuntos de treino o erro quadrático é expresso como:

$$E = \sum_{k=1}^P E_k = \sum_{k=1}^P \sum_{i=1}^{N_m} \left[\frac{1}{2} (x_i^k - x_i)^2 \right]_k \quad (4.6)$$

A variação no peso $\Delta w_{ij}^{(m)}$ é a variação feita no peso $w_{ij}^{(m)}$ entre o neurônio j da camada m e o neurônio i da camada $(m + 1)$ e η uma constante denominada coeficiente de aprendizagem a cada ajuste e é expressa como:

$$\Delta w_{ij}^{(m)} = -\eta \frac{\partial E}{\partial w_{ij}^{(m)}}, \quad \eta > 0 \quad (4.7)$$

Derivando a função erro em função do peso, tem-se:

$$\frac{\partial E}{\partial w_{ij}^{(m)}} = \frac{\partial E}{\partial x_i^{(m+1)}} \frac{\partial x_i^{(m+1)}}{\partial w_{ij}^{(m)}} \quad (4.8)$$

Baseado na redução do erro quadrático médio, utilizando-se do cálculo da derivada parcial de E em relação a cada peso, ou seja, o declínio do gradiente de E foi criado o algoritmo *backpropagation*, também conhecido por *backpropagation* Gradiente Descendente.

Da expressão (4.4), ao derivar parcialmente a saída do neurônio i da camada $(m + 1)$ em relação ao peso, utilizando a regra da função composta chega-se a:

$$\begin{aligned} \frac{\partial x_i^{(m+1)}}{\partial w_{ij}^{(m)}} &= \frac{\partial f_{(m+1)}(a_i^{(m+1)})}{\partial w_{ij}^{(m)}} = \frac{\partial f_{(m+1)}(a_i^{(m+1)})}{\partial a_i^{(m+1)}} \frac{\partial a_i^{(m+1)}}{\partial w_{ij}^{(m)}} \\ &= f'_{m+1} \frac{\partial a_i^{(m+1)}}{\partial w_{ij}^{(m)}} \end{aligned} \quad (4.9)$$

Considerando-se (4.2)

$$\begin{aligned} \frac{\partial x_i^{(m+1)}}{\partial w_{ij}^{(m)}} &= f'_{m+1} \frac{\partial a_i^{(m+1)}}{\partial w_{ij}^{(m)}} = f'_{m+1} \frac{\partial}{\partial w_{ij}^{(m)}} \left(\sum_{j=0}^{N_m} w_{ij}^{(m)} x_j^{(m)} \right) \\ &= f'_{m+1} x_j^{(m)} \end{aligned} \quad (4.10)$$

Em relação ao outro termo da derivada composta da equação (4.8), considera-se, para ajuste dos pesos entre a camada $M - 1$ e a camada de saída M , a definição de erro:

$$\frac{\partial E}{\partial x_i^{(M)}} = \frac{\partial}{\partial w_{ij}^{(m)}} \left[\frac{1}{2} \sum_{i=1}^{N_m} (x_i^d - x_i^{(M)})^2 \right] = -(x_i^d - x_i^{(M)}) \quad (4.11)$$

sendo o incremento de peso obtido pela expressão

$$\Delta w_{ij}^{(M-1)} = -\eta \alpha_i^{(M)} x_j^{(M-1)} \quad (4.12)$$

Nesta regra de ajuste dos pesos $\alpha_i^{(M)}$ é denominado sinal de erro generalizado, expresso por:

$$\alpha_i^{(M)} = -(x_i^d - x_i^{(M)}) f'_M \quad (4.13)$$

De forma a obter o ajuste dos pesos entre as camadas intermediárias e partindo da definição de erro, obtém-se por derivação:

$$\frac{\partial E}{\partial x_i^{(M-1)}} = \sum_{n=1}^{N_M} \frac{\partial E}{\partial x_n^{(M)}} \frac{\partial x_n^{(M)}}{\partial x_i^{(M-1)}} \quad (4.14)$$

Considerando (4.2) e (4.14) para o primeiro e segundo termos do somatório respetivamente, chega-se a:

$$\begin{aligned} \frac{\partial x_n^{(M)}}{\partial x_i^{(M-1)}} &= \frac{\partial f_{(M)}(a_n^{(M)})}{\partial a_n^{(M)}} \frac{\partial a_n^{(M)}}{\partial x_i^{(M-1)}} \\ &= f'_M \frac{\partial}{\partial x_j^{(M)}} \sum_{j=0}^{N_{M-1}} w_{nj}^{(M-1)} x_j^{(M-1)} = f'_M w_{ni}^{(M-1)} \end{aligned} \quad (4.15)$$

Em relação à x_i a equação (4.14) torna-se:

$$\frac{\partial E}{\partial x_i^{(M-1)}} = \sum_{n=1}^{N_M} \left[-(x_i^d - x_i^{(M)}) f'_M w_{ni}^{(M-1)} \right] = \sum_{n=1}^{N_M} \alpha_n^{(M)} w_{ni}^{(M-1)} \quad (4.16)$$

Considerando a expressão desta derivada na equação, tem-se:

$$\frac{\partial E}{\partial x_i^{(M-2)}} = \frac{\partial E}{\partial x_i^{(M-2)}} \frac{\partial x_i^{(M-1)}}{\partial w_{ij}^{(M-2)}} = f'_{M-1} x_j^{(M-2)} \sum_{n=1}^{N_M} \alpha_n^{(M)} w_{ni}^{(M-1)} \quad (4.17)$$

Logo, o incremento para os pesos entre as camadas intermediárias é expresso por:

$$\Delta w_{ij}^{(M-2)} = -\eta \alpha_i^{(M-1)} x_j^{(M-2)} \quad (4.18)$$

Em que o erro generalizado das camadas intermediárias $\alpha_i^{(M-1)}$ é obtido por:

$$\alpha_i^{(M-1)} = f'_{M-1} \sum_{n=1}^{N_M} \alpha_n^{(M)} w_{ni}^{(M-1)} \quad (4.19)$$

Em redes com várias camadas intermediárias, para cada camada m é necessário calcular $\alpha_i^{(m)}$.

Generalizando o ajuste, a equação (4.18) pode ser reescrita como:

$$\Delta w_{ij}^{(m)} = -\eta \alpha_i^{(m+1)} x_j^{(m)} \quad (4.20)$$

Em que $\alpha_i^{(m+1)}$ se obtém por

$$\alpha_i^{(m+1)} = f'_{m+1} \sum_{n=1}^{N_{m+2}} \alpha_n^{(m+2)} w_{ni}^{(m+1)} \quad (4.21)$$

Os valores dos pesos podem ser corrigidos por padrão ou por época. Isso implica que o treinamento pode ocorrer por entrada sequencial de pares de entrada e saída para o caso de correção por padrão, ou por apresentação de todos os padrões em lotes de dados quando a correção é por época. A cada iteração do algoritmo, ou seja, a cada ajuste nos valores dos pesos, obtém-se um incremento Δw , adicionado ao vetor w obtido na interação anterior. Além disso, é um método de primeira ordem, portanto, utiliza somente a informação das derivadas de primeira ordem do erro quadrático de aproximação para correção dos pesos.

Este algoritmo tem como vantagem a sua simplicidade, permitindo que seja implementado rapidamente. A morosidade da sua convergência torna-se uma

desvantagem à sua aplicação, e pode ser piorada caso os parâmetros da rede forem configurados inadequadamente (Almeida, 2012).

Deve-se considerar ainda que a superfície do erro de resposta pode não ser tão simples, e dificulta a obtenção das derivadas do erro para posterior ajuste dos pesos. Assim, as redes ficam sujeitas aos problemas de mínimos locais na superfície de erro e resultam em falsa convergência do ajuste. Caso seja encontrado um mínimo local, o erro para o conjunto de treinamento para de diminuir e estaciona em um valor maior que o aceitável. Todos esses fatores tornam o algoritmo inviável, pelo ponto de vista de desempenho computacional.

4.10 Treinamento Rápido

Para permitir uma utilização mais ampla, muitas técnicas têm sido discutidas para acelerar o desempenho do *backpropagation*, de modo a oferece maior flexibilidade e uma avaliação computacional mais rápida. Existem vários algoritmos de treinamento de alto desempenho que podem convergir mais rapidamente em relação ao algoritmo *backpropagation* gradiente descendente. São divididos em duas categorias: os que utilizam técnicas heurísticas desenvolvidas a partir do algoritmo descendente padrão, como a utilização do termo *momentum* por exemplo. E a segunda categoria, que utiliza técnicas de otimização numérica tais como gradiente conjugado, Quasi-Newton e Levenberg-Marquardt.

4.10.1 Método de Levenberg-Marquardt

Levenberg-Marquardt propôs um algoritmo baseado no método de otimização de Newton que consiste na determinação das derivadas de segunda ordem do erro quadrático em relação aos pesos, para aceleração do treinamento. O algoritmo proposto acrescenta restrição adicional $\mu I(w)$, onde μ é uma constante real positiva e I uma matriz identidade com dimensões apropriadas à matriz $[J^T(w)J(w)]$. Desta forma, para a iteração $(n + 1)$, a correção do peso é dada como:

$$\Delta w(n) = -[J^T(w)J(w) + \mu(n)I]^{-1}[J^T(w(n))][e(w(n))] \quad (4.22)$$

O método é dito mais rápido porque μ é ajustado com um fator variante (maior ou menor que 1) de acordo com o resultado da função erro para cada iteração, o que pode conduzir a convergência mais rapidamente. No entanto, demanda mais cálculos por iteração, a depender da complexidade das funções de ativação e do problema a ser analisado. De acordo com Saini e Soni (2002) a utilização torna-se pouco prática quando se trabalha com número elevado de iterações. E apesar do grande esforço computacional no sentido de se trabalhar a formulação, ele segue sendo o algoritmo de treinamento mais rápido para Redes Neurais.

Além disso, o método é especialmente apropriado para problemas mal condicionados.

4.11 Eficiência de uma Rede Neural

Normalmente a escolha da configuração mais apropriada de uma Rede Neural Artificial para melhor representação do domínio do problema ocorre de forma empírica, embora já existam heurísticas de estimativa de topologias de melhor desempenho. A seguir são discutidos alguns aspectos que estão diretamente relacionados a eficiência das aplicações que utilizam rede neural como ferramenta.

Pré-processamento de dados: em muitos casos os padrões apresentados à rede para treinamento apresentam discrepância de intervalo ou naturezas diferentes. Pode fazer-se necessário, nesses casos, a adoção de técnicas que minimizem essas características dos dados. A normalização é uma dessas técnicas e tem sido empregada por aumentar a eficiência do algoritmo, uma vez que elimina o efeito da dimensão e assegura que todas as variáveis recebam a mesma atenção durante o ajuste dos pesos. Logo, o processo de normalização de padrões tende a minimizar os problemas advindos do uso de unidades distintas entre as variáveis, além de evitar resultados equivocados a partir de dados errôneos e não representativos. A normalização também pode evitar estagnação da rede e pode acelerar a velocidade de convergência. Além da normalização existem outras técnicas para pré-processamento de dados como análise de componentes

principais (PCA), entre outras.

Número de neurônios de entrada e saída: na camada de entrada recomenda-se que o número de neurônios de entrada seja igual ao número de variáveis que produzem as entradas da rede. Para determinação do número de neurônios na camada de saída, deve-se considerar a pretensão da rede. De acordo com Braga et. Al (1999) em problemas de classificação de itens em agrupamentos o recomendável é que haja um neurônio para cada grupo. Para modelagem de funções matemáticas, um neurônio é suficiente. Quanto a filtragem de ruídos um número de neurônios na camada de saída igual ao da camada de entrada já é satisfatório.

Número de neurônios de camadas intermediárias: para Braga et. Al (1999) a determinação do número de neurônios nas camadas intermediárias também é uma questão relevante no projeto da arquitetura de uma Rede Neural. Quando esse número é subestimado pode ocorrer o chamado *underfitting*. Neste caso a rede apresenta dificuldades em manipular as entradas por apresentar estrutura insuficiente à demanda, o que pode dificultar a convergência dos algoritmos de treinamento. Em contrapartida, um número excessivo de neurônios na camada intermediária diminui a capacidade de generalização, o que caracteriza o *overfitting*. Hecht-Nielsen (1989) propõe que a camada oculta tenha aproximadamente $(2i + 1)$ neurônios, onde i é o número de variáveis de entrada. Braga *et al.* (1999) sugere que a quantidade de neurônios da camada oculta seja um número intermediário entre o número de neurônios na camada de entrada e o número de neurônios na camada de saída. Ou ainda $2/3$ do número de neurônios na entrada mais o número de neurônios na saída. Também se adota número de neurônios da camada intermediária como 10% do número de exemplos. Almeida (2012) conclui que a partir de certo número de neurônios, a rede passa a estar bem aproximada no geral, dado que o erro relativo médio não sofre grandes alterações. Mas em certos pontos esta aproximação já não é muito exata, demonstra que o aumento do número de neurônios da rede aproxima melhor dos valores do conjunto de treinamento, tendo mais dificuldade em aproximar valores que não façam parte deste conjunto.

Critérios de Parada: os métodos que determinam o momento em que o treinamento deve ser interrompido são chamados de critérios de parada. Os mais utilizados segundo Haykin (1999) são:

- Número máximo de ciclos, conhecido também como iterações ou *epochs*:

número de vezes que o conjunto de padrões é apresentado à rede e os pesos consequentemente ajustados;

- Norma euclidiana do vetor gradiente descendente de valor menor que o gradiente limite (algoritmo *backpropagation*);
- Taxa absoluta do erro quadrático médio muito pequena (algoritmo *backpropagation*);
- Tempo limite para iterações; e
- Critérios de parada híbridos os quais se estabelecem formulações que relacionam os critérios anteriormente citados.

5 TÉCNICA PROPOSTA APLICADA A ANÁLISE DE ESTRUTURAS

Formulações em métodos numéricos são amplamente difundidas na análise de estruturas. A compreensão de fenômenos físicos em engenharia estrutural, normalmente, é feita aplicando-se esses métodos e estabelecendo-se relações matemáticas que transcrevam a fenomenologia na forma de comportamento do material, condições de vinculação e carregamento e arranjo estrutural. Essas relações podem ser descritas pela aproximação de função matemática que descreve o comportamento de grandezas de interesse na análise e as quais podem implicar em custo computacional e de desenvolvimento.

Este trabalho apresenta possíveis aplicações onde a modelagem RNA pode ser integrada ao Método dos Elementos Finitos. De maneira geral, pretende-se aproximar curvas que descrevem determinadas relações importantes à análise estrutural por meio da operação de Redes Neurais treinadas com padrões obtidos de simulações em MEF.

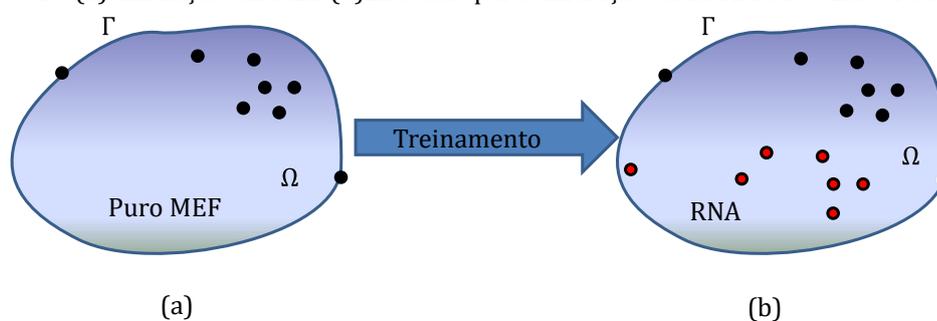
5.1 Metodologia

Baseada na literatura já citada a metodologia do presente trabalho evoluiu no sentido de explorar e testar a aplicabilidade de Redes Neurais a problemas característicos da engenharia estrutural. Inicialmente, para efeitos de validação, foram abordados exemplos cuja solução apresenta-se matematicamente exata, de complexidade conhecida. As Redes Neurais treinadas a partir de dados extraídos de soluções analíticas objetivaram aproximar as curvas destas funções matemáticas. Nesta fase do trabalho foi elaborado um estudo acerca das melhores configurações para ajustar parâmetros de treinamento que pudessem tornar a rede mais eficiente em fase de operação. Verificou-se também a influência desses parâmetros na qualidade de resposta fornecida pela RNA e estabeleceu-se os critérios de configuração de tais parâmetros para a modelagem de problemas de complexidade elevada.

Posteriormente combinou-se o uso de Redes Neurais com o Método dos Elementos Finitos. Nesta junção utilizou-se da modelagem MEF para a geração de padrões de treinamento para a RNA, em duas linhas de aplicação:

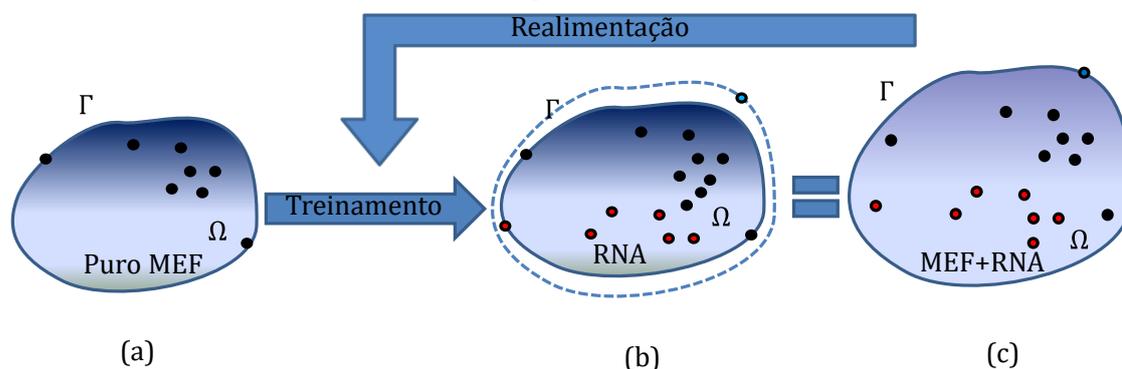
- Interpolação: a partir do treinamento de uma RNA com dados em MEF foram simulados valores de grandezas cujos valores de entrada estão contidos entre os intervalos de máximo e mínimo do conjunto de treinamento, ou seja, dados contidos dentro do domínio Ω do problema, análogo ao procedimento de interpolação. A solução final contempla solução em MEF e por Redes Neurais, como ilustra a Figura 5. 1.

Figura 5. 1 – (a) Simulação em MEF (b) Entradas para simulação da RNA. Fonte: Elaborada pela autora



- Extrapolação: Uma rede foi treinada e em fase de operação é simulada com valores contidos na fronteira Γ do conjunto de treinamento, Figura 5. 2 (a). A saída correspondente realimenta o conjunto de treinamento usado durante o treinamento anterior e os pesos reajustados. Este processo é repetido continuamente até que a curva a qual se deseja estimar seja totalmente descrita. Com isso, a fronteira que delimita o conjunto de entrada é também continuamente expandida a cada novo valor extrapolado. Pode-se fazer necessário, em caso de pequena quantidade de amostras obtidas via MEF puro, utilizar de interpolação para aumentar o conjunto de dados de treinamento. A Figura 5. 2 exhibe este processo.

Figura 5. 2 – (a)Puro MEF (b)Entradas para simulação RNA (c) Solução MEF+RNA obtida por simulação com pares de entrada em puro MEF, por interpolação e por extrapolação. Fonte: Elaborada pela autora



Todos os códigos, elementos finitos e Redes Neurais, foram desenvolvidos no ambiente MATLAB®, em computador com processador Intel Core i3 2,67GHz e 3 GB de RAM. As configurações dos parâmetros de treinamento da RNA, para cada exemplo, foram obtidas através de comparativo de desempenho entre operação da rede treinada com topologias mais simples até as mais complexas. Foi mantido o algoritmo *default* de treinamento do ambiente, o *backpropagation* Levenberg-Marquardt. Para ilustrar a metodologia proposta, foram abordados problemas de elasticidade plana, considerando não linearidade geométrica. A solução generalizada proposta através de MEF+RNA foi aplicada a três classes de problemas como será discutido posteriormente.

5.2 Problemas com Solução Analítica Conhecida

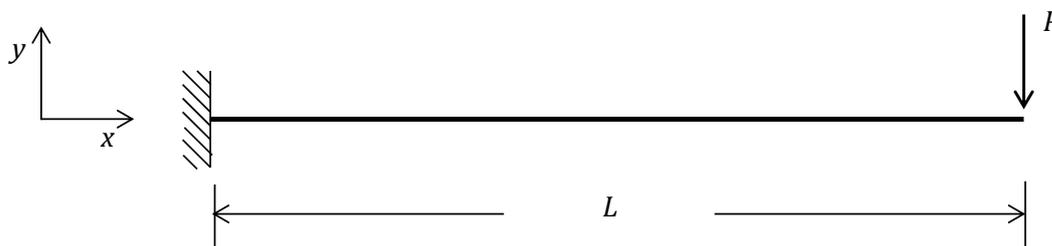
O objetivo deste tópico é fazer análises preliminares acerca de configurações da Rede Neural Artificial que impliquem em aumento de desempenho durante treinamento e operação. São apresentados exemplos cuja solução apresenta-se exata, de complexidade conhecida, e feitos testes variando-se os parâmetros de treinamento da rede, tais como função de ativação, número de amostras e número de camadas. Este procedimento visa estabelecer critérios para configuração inicial da RNA nos problemas mais complexos que serão abordados posteriormente.

5.2.1 Viga em balanço sob carga pontual

Considere-se o modelo unidimensional de viga engastada-livre, com comprimento $L = 1$ m, ilustrado na Figura 5. 3. Admita-se seção retangular medindo 15 x 25 cm, em

concreto armado de módulo de elasticidade $E = 25 \text{ GPa}$.

Figura 5.3 - Modelo unidimensional de viga sob ação de carga pontual. Fonte: Elaborada pela autora.

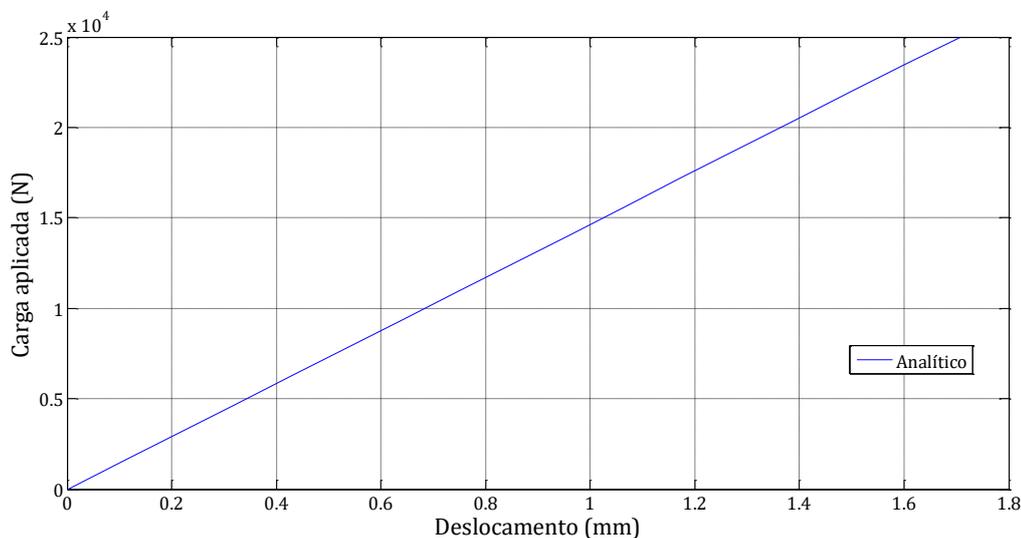


O deslocamento u na extremidade livre da viga sob a ação de uma carga pontual P é descrito como:

$$u = \frac{PL^3}{3EI} \quad (5.1)$$

Ao variar o valor de carga aplicada entre 0 a 25 kN obtém-se o comportamento dos deslocamentos sofridos pela viga, em módulo, como mostra a Figura 5.4.

Figura 5.4 - Solução analítica da flecha máxima na viga sob ação de carga pontual.



Para modelagem da solução da viga por Redes Neurais foi utilizada a arquitetura *Perceptron* e o algoritmo de treinamento *backpropagation*. O *Perceptron* é a mais simples dentre as arquiteturas disponíveis e consiste apenas em variáveis de entrada e neurônios correspondentes às variáveis de saída da rede. Segundo a literatura, sua aplicação está limitada à modelagem onde a solução é caracterizada como linearmente separável. A entrada da RNA foi a carga P e a saída o deslocamento correspondente à carga aplicada, sendo portanto a arquitetura de neurônios 1:1.

Amostras para a rede foram extraídas a partir da equação analítica para o treinamento de 5 redes com 2, 3, 5, 9 e 17 pares de entrada e saída. Esta sequência de amostragem garante que contenham dados comuns aos padrões das 5 redes treinadas, de modo a facilitar a comparação de resultados advindos de simulação. Assim foi possível verificar a influência da quantidade de amostras utilizadas para treinamento na qualidade da resposta. A simulação das 5 redes com 17 amostras treinadas com as funções de ativação linear, logística e tangente hiperbólica apresentou os seguintes resultados, ilustrados nas Figuras 5.5 a 5.7.

Figura 5.5 – Dados analíticos e resultados da RNA treinada com diferentes quantidades de amostras e função de ativação linear.

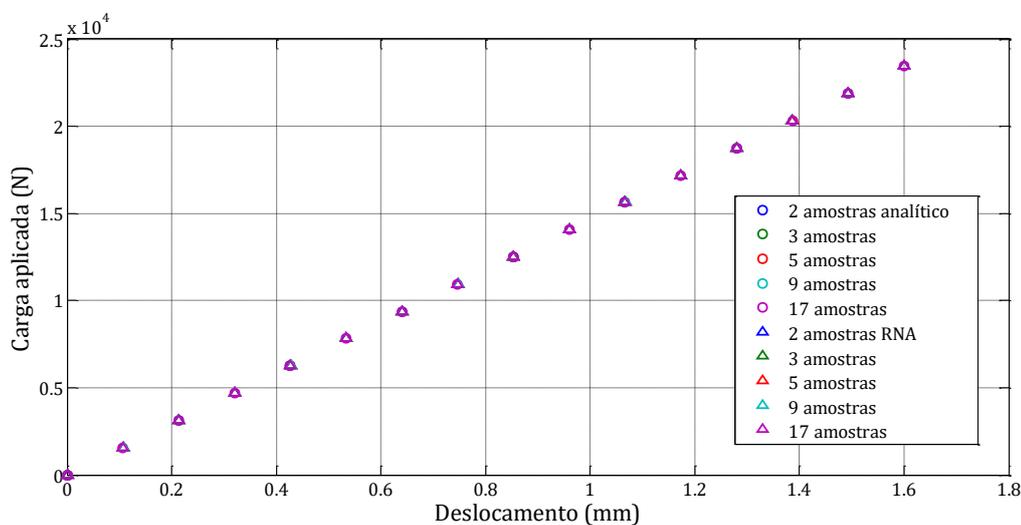


Figura 5.6 - Dados analíticos e resultados da RNA treinada com diferentes quantidades de amostras e função de ativação logística.

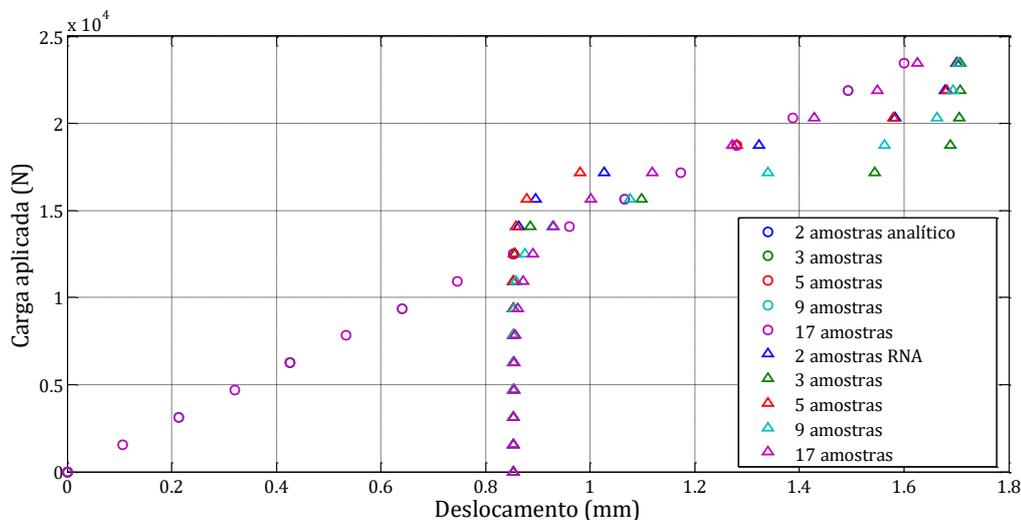
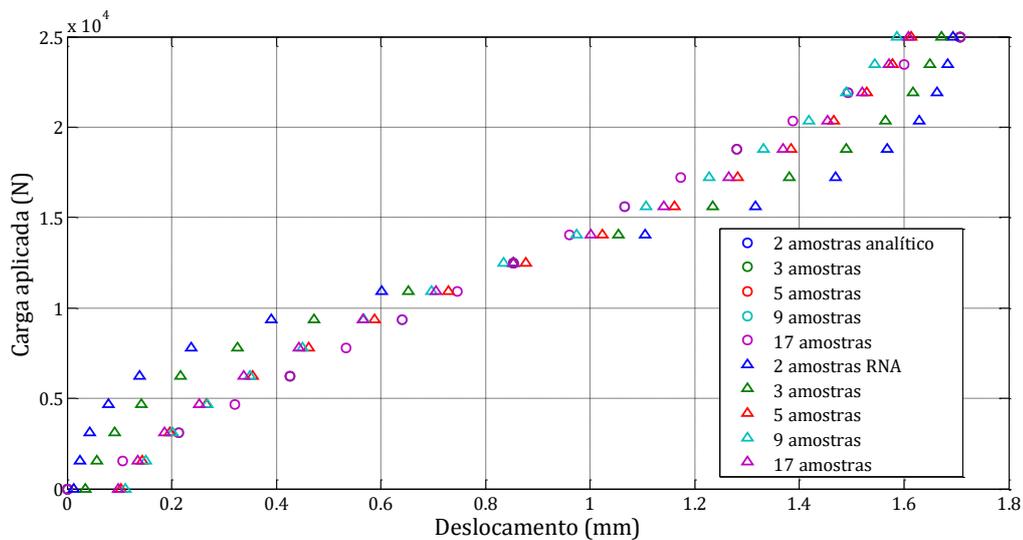


Figura 5.7 - Dados analíticos e resultados da RNA treinada com diferentes quantidades de amostras e função de ativação tangente hiperbólica.



Os erros nos resultados obtidos por RNA para a carga aplicada em relação à solução analítica para rede treinada com função de ativação linear, logística e tangente hiperbólica são apresentados nas Figuras 5.8 a 5.10.

Figura 5.8 – Erros relativos entre resultado analítico e por RNA treinada com diferentes quantidades de amostras e função de ativação linear.

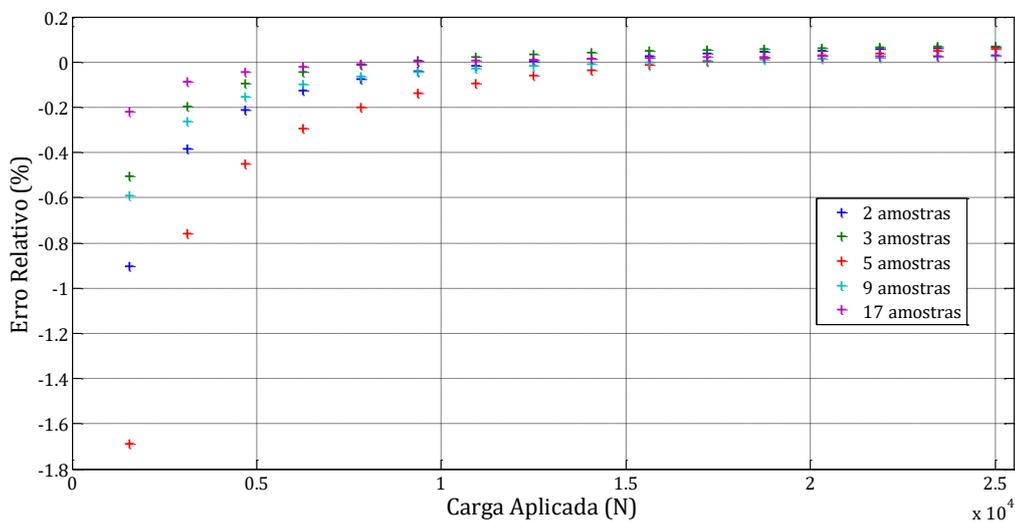


Figura 5. 9 - Erros relativos entre resultado analítico e por RNA treinada com diferentes quantidades de amostras e função de ativação logística.

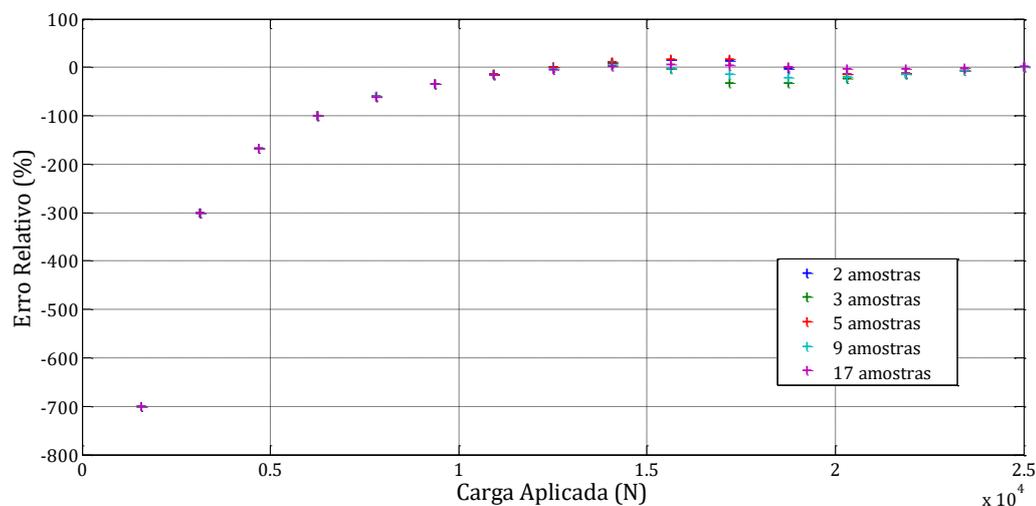
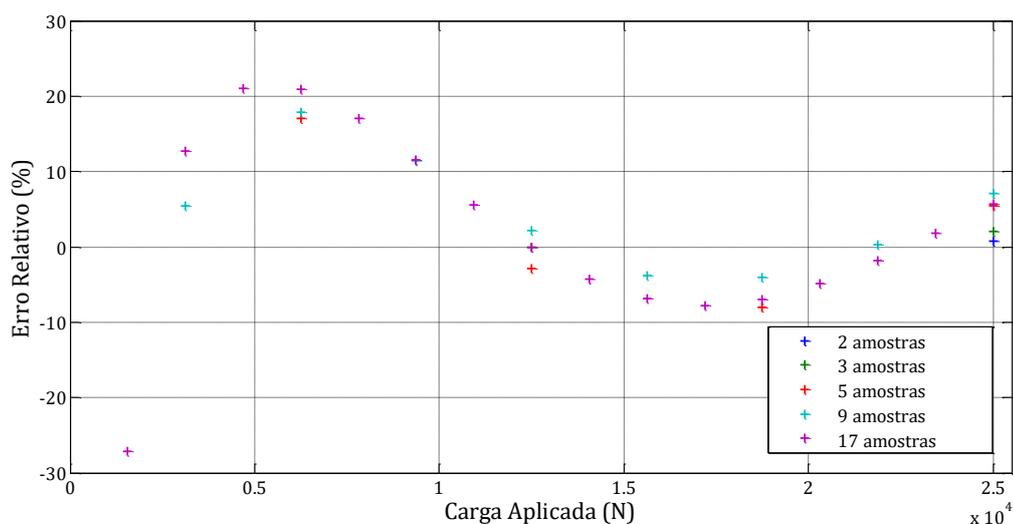


Figura 5. 10 - Erros relativos entre resultado analítico e por RNA treinada com diferentes quantidades de amostras e função de ativação tangente hiperbólica.



De acordo com os gráficos da Figura 5. 8 os erros relativos referentes aos resultados obtidos por treinamento com função de ativação linear apresentaram-se menores quando comparados aos produzidos pelo treinamento com as demais funções, Figura 5. 9 e Figura 5. 10, independentemente do tamanho da amostra.

Também é possível observar que os erros entre previsto e analítico são menores à medida que a carga prevista é aumentada. Isso ocorre devido à quantidade e características das amostras de treinamento, que podem ter sido insuficientes para descrever o domínio do problema com qualidade.

Portanto, como esperado, conhecida a complexidade da função para aproximação

que estabelece uma relação linear entre a ação da carga e a flecha na viga, uma RNA tipo *Perceptron* treinada com função de ativação linear apresenta bom desempenho. Para a arquitetura de rede 1:1, nota-se que funções de ativação mais complexas, associadas a amostras de treinamento pequenas, podem levar a erros significativos, o que pode inviabilizar sua aplicação. Além disso, pode-se ressaltar que a RNA, para este problema, poderia também convergir com demais arquiteturas, com diferentes quantidades de camadas e combinações de funções de ativação. Entretanto, configurações mais complexas demandam mais tempo de processamento, uma vez que o algoritmo trabalha com maior número de variáveis.

5.2.2 Modelo constitutivo de Popovics (1973)

As leis constitutivas definem o comportamento do material quando submetido a solicitações, através do diagrama tensão-deformação. Para o concreto sujeito à compressão, Popovics (1973) propõe um modelo elástico não linear descrito por:

$$\sigma = \sigma_0 \left(\frac{\varepsilon}{\varepsilon_0} \right) \frac{m}{m - 1 + \left(\frac{\varepsilon}{\varepsilon_0} \right)^m} , \quad (5.2)$$

Onde σ é a tensão de compressão após o pico de resistência, σ_0 é a resistência à compressão do concreto, ε a deformação atual, ε_0 a deformação específica correspondente à tensão de resistência à compressão e m um parâmetro dependente de σ_0 , definido por $m = 4 \times 10^{-4} \sigma_0 + 1$, em psi. Para equivalência com o sistema métrico, valores de m entre 2 e 5 correspondem, aproximadamente, a concretos com resistência à compressão entre 20 Mpa e 70 MPa.

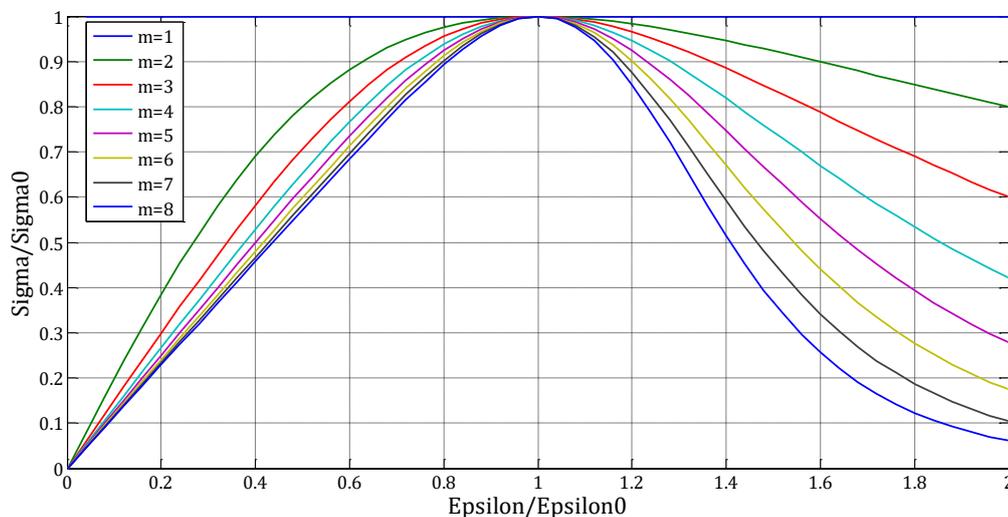
Reescrevendo (5.2) de forma alternativa:

$$\frac{\sigma}{\sigma_0} = \left(\frac{\varepsilon}{\varepsilon_0} \right) \frac{m}{m - 1 + \left(\frac{\varepsilon}{\varepsilon_0} \right)^m} . \quad (5.3)$$

A curva que descreve a equação (5.3) é ilustrada na

Figura 5. 11.

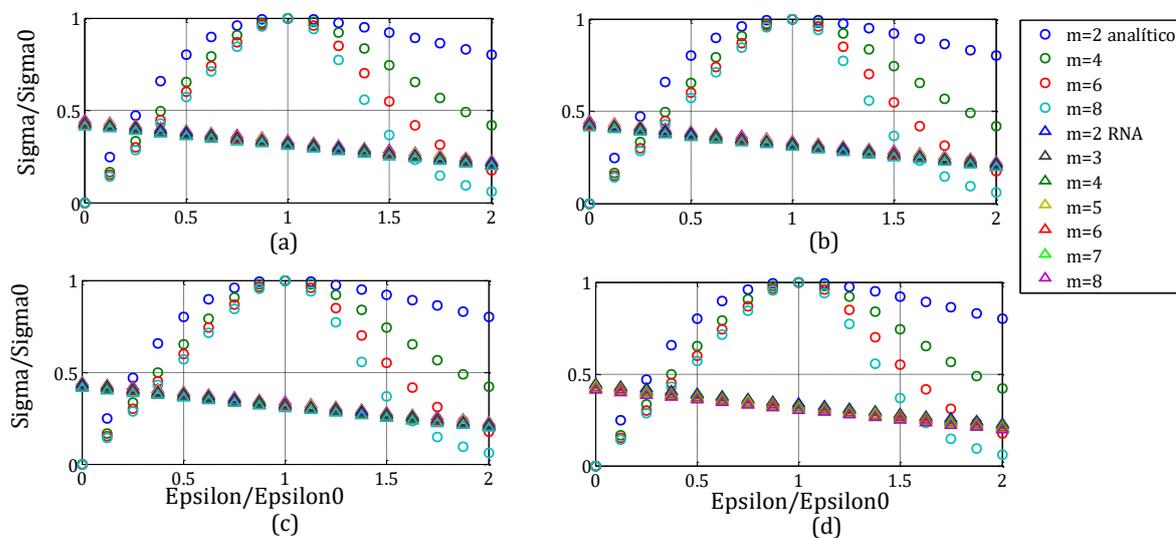
Figura 5.11 – Modelo constitutivo elástico não linear Popovics (1973) para o concreto à compressão.



A modelagem deste problema por RNA deu-se através de treino e simulação partindo do modelo mais simples de rede existente até um modelo mais complexo. Foram variáveis de entrada $\varepsilon/\varepsilon_0$ e m e variável de saída σ/σ_0 , em 4 redes tipo *Perceptron*, topologia 2:1, cada qual com 3, 5, 9 e 17 conjuntos de treinamento, respectivamente. Cada rede foi simulada com 17 dados nas entradas, com $\varepsilon/\varepsilon_0$ variando de 0 a 2 e m com valores 2, 4, 6 e 8.

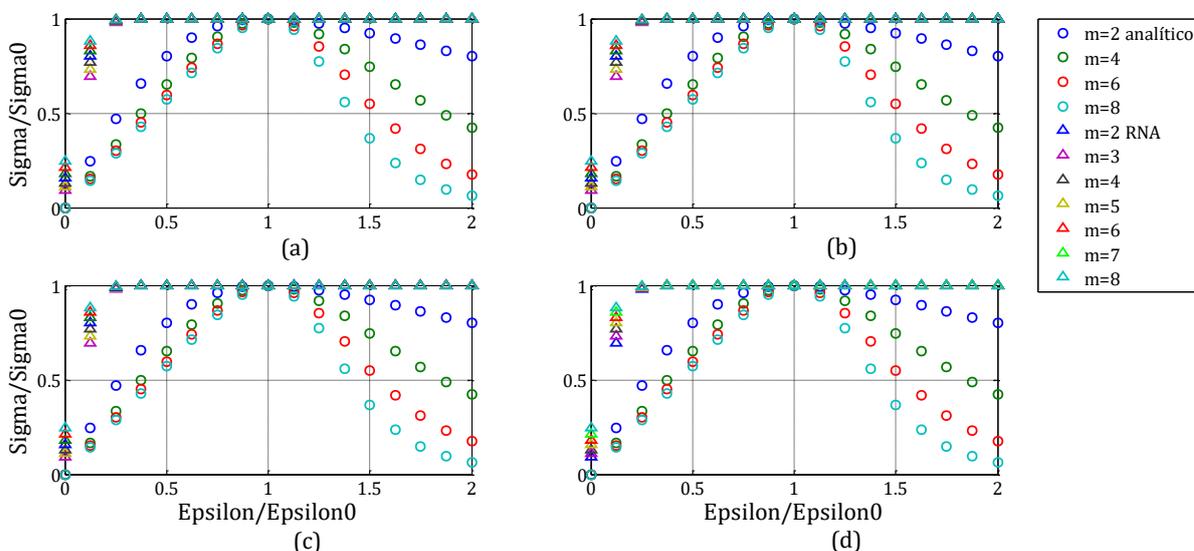
Para treinamento com função de ativação linear têm-se os gráficos da Figura 5.12.

Figura 5.12 – Simulação de *Perceptron* treinado com função de ativação linear com (a) 3 amostras (b) 5 amostras (c) 9 amostras (d) 17 amostras.



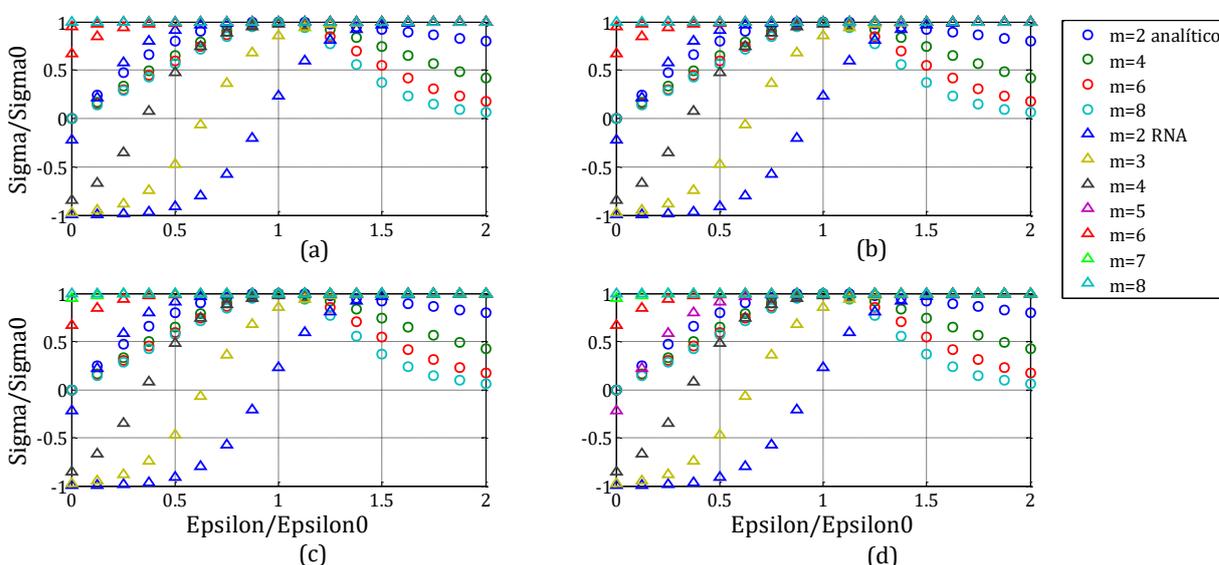
Para função de ativação logística, os resultados são ilustrados na Figura 5.13.

Figura 5. 13 - Simulação de *Perceptron* treinado com função de ativação logística com (a) 3 amostras (b) 5 amostras (c) 9 amostras (d) 17 amostras.



Por fim, para função tangente hiperbólica, têm-se os resultados da Figura 5.14.

Figura 5. 14 - Simulação de *Perceptron* treinado com função de ativação tangente hiperbólica com (a) 3 amostras (b) 5 amostras (c) 9 amostras (d) 17 amostras.

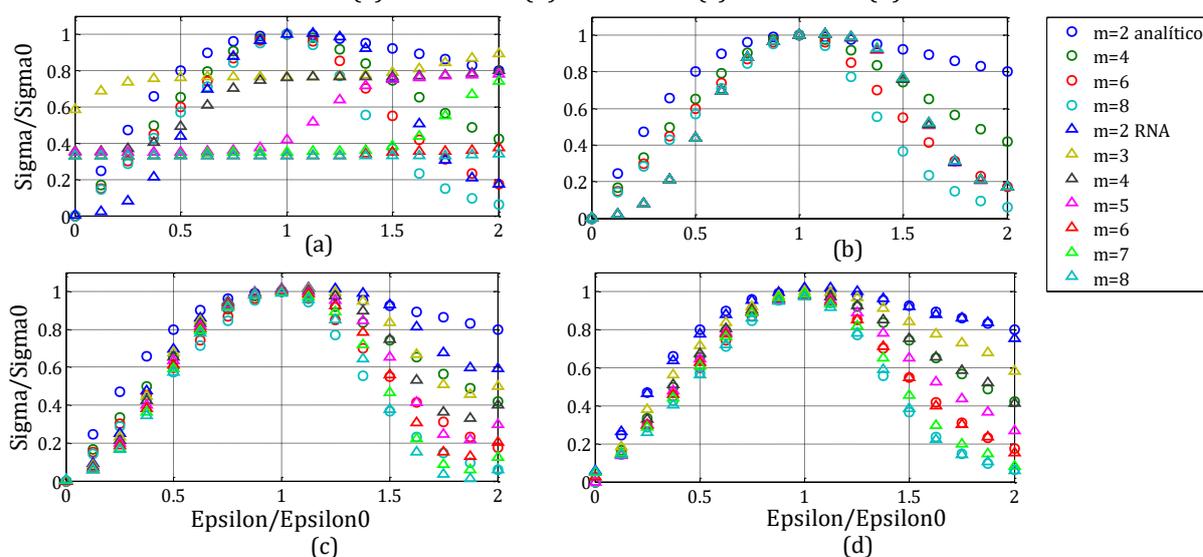


Com base nos resultados da Figura 5. 12, Figura 5. 13 e Figura 5. 14 pode-se constatar que o *Perceptron* não se adequa à modelagem deste problema e apresenta discrepância entre os resultados simulados e os analíticos para todas as possibilidades de treino testadas. Tornou-se, então, necessária a utilização de uma rede multicamadas (MLP). A incapacidade da função de ativação linear em representar o fenômeno era esperada, sendo mostrada de forma didática, por se tratar de um exemplo ilustrativo das

potencialidades da modelagem via RNA em problemas não lineares.

Para uma rede MLP é preciso inserir camadas intermediárias entre a entrada e a camada de saída. Inicialmente, testou-se a configuração com uma única camada intermediária, do tipo 2:X:1. Para a camada intermediária e camada de saída foram feitas as seguintes combinações entre as funções de ativação: linear-linear, logística-logística, tangente hiperbólica-tangente hiperbólica, logística-linear, tangente hiperbólica-linear, linear-logística, tangente hiperbólica-logística, linear-tangente hiperbólica, logística-tangente hiperbólica. Ao incrementar o número de neurônios da camada intermediária de 1 a 10 e seguindo a ordem de funções de ativação citadas acima os resultados não expressaram convergência significativa. Configurações de rede que continham entre 2 e 6 neurônios na camada intermediária e funções de ativação logística-linear, treinadas com 16 amostras, representaram adequadamente o formato das curvas analíticas, o que não ocorreu nas demais configurações. No entanto, esta topologia de rede levou a erros consideráveis em relação aos valores da solução analítica em diversos pontos, tornando-a inviável. Isso justificou, portanto, a inserção de uma nova camada intermediária. Para inserção desta nova camada foi considerado que a camada de saída deveria permanecer com função de ativação linear, *default* do Matlab. Assim a combinação entre as funções de ativação usuais deu-se apenas entre as 2 camadas intermediárias.

Figura 5. 15 – Simulação de MLP 2:4: 4:1 treinada com função de ativação linear e logística nas camadas intermediárias (a) 3 amostras (b) 5 amostras (c) 9 amostras (d) 17 amostras.



Após treinamento realizado com combinação entre funções de ativação e variação no número de neurônios chegou-se a topologia 2:4:4:1, função de ativação linear e

logística nas camadas intermediárias e treinamento a partir de 16 combinações de entrada como a topologia de rede que forneceu os melhores resultados, como ilustra a Figura 5. 15.

A Tabela 5. 1 apresenta a solução via RNA e solução analítica para o modelo de Popovics(1973) em 17 pontos de deformação para diferentes valores de m .

Tabela 5. 1 - Comparativo de resultados RNA e analítico.

$\varepsilon/\varepsilon_0$	σ/σ_0							
	$m = 2$		$m = 4$		$m = 6$		$m = 8$	
	Analítico	RNA	Analítico	RNA	Analítico	RNA	Analítico	RNA
0	0	0,056	0	0,022	0	0,0273	0	0,052
0,125	0,246	0,264	0,166	0,144	0,150	0,1469	0,142	0,141
0,250	0,470	0,462	0,332	0,327	0,300	0,2949	0,285	0,258
0,375	0,657	0,635	0,496	0,509	0,449	0,4609	0,428	0,401
0,500	0,800	0,774	0,653	0,672	0,598	0,6281	0,571	0,561
0,625	0,898	0,878	0,793	0,804	0,741	0,7773	0,711	0,722
0,750	0,960	0,951	0,904	0,900	0,869	0,8922	0,845	0,859
0,875	0,991	0,996	0,976	0,959	0,963	0,9614	0,953	0,948
1,000	1,000	1,018	1,000	0,983	1,000	0,9789	1,000	0,971
1,125	0,993	1,019	0,977	0,971	0,960	0,9419	0,940	0,915
1,250	0,975	1,000	0,918	0,926	0,850	0,8508	0,771	0,781
1,375	0,951	0,964	0,836	0,849	0,701	0,7125	0,556	0,589
1,500	0,923	0,918	0,744	0,751	0,549	0,5487	0,367	0,383
1,625	0,892	0,878	0,651	0,655	0,416	0,4002	0,233	0,222
1,750	0,861	0,858	0,565	0,582	0,311	0,3016	0,147	0,142
1,875	0,830	0,837	0,488	0,517	0,232	0,2349	0,093	0,107
2,000	0,800	0,753	0,421	0,413	0,173	0,1535	0,060	0,060

Os erros relativos entre a solução analítica e simulação RNA são descritos na Tabela 5. 2.

Tabela 5. 2 - Erros relativos entre resultados por RNA e analítico.

$\varepsilon/\varepsilon_0$	Erros relativos(%)			
	$m = 2$	$m = 4$	$m = 6$	$m = 8$
0	-	-	-	-
0,125	7,554	13,617	2,066	1,259
0,250	1,678	1,682	1,7	9,695
0,375	3,422	2,496	2,490	6,347
0,500	3,237	2,939	5,015	1,610
0,625	2,258	1,475	4,870	1,446
0,750	0,895	0,420	2,657	1,668
0,875	0,554	1,659	0,217	0,461
1,000	1,85	1,69	2,11	2,87
1,125	2,638	0,623	1,936	2,720
1,250	2,572	0,837	0,011	1,308
1,375	1,408	1,589	1,539	6,040
1,500	0,476	1,007	0,072	4,214
1,625	1,601	0,506	3,890	4,664
1,750	0,313	3,076	3,147	3,188
1,875	0,842	6,000	1,162	14,164
2,000	5,762	1,899	11,730	0,657

Observa-se na Tabela 5.2 erros relativamente baixos, sendo o maior erro percentual obtido de 14,164%. Este erro quando analisado isoladamente pode indicar um mau desempenho da topologia e configurações de RNA. Entretanto, dentro do conjunto de resultados fornecidos na Tabela 5.1 e Tabela 5.2, observa-se convergência nas respostas entre RNA e analítico, com predominância de erros bem abaixo do máximo encontrado.

Conclui-se que as Redes Neurais com múltiplas camadas mostraram-se eficazes para simular o problema. Conhecido ε_0 e σ_0 , referentes à propriedade de resistência do material, é possível determinar a tensão para qualquer nível de deformação, dado o valor de m , parametrizando os dados. Portanto, os valores expressos na Tabela 5. 1 e 5.2 referentes à resposta por RNA evidenciam boa aproximação e caracterizam a curva como

pretendido.

5.3 Problemas com Solução Aproximada via Formulação Linear do MEF

Uma das maneiras de se obter os deslocamentos em diferentes pontos de determinada geometria é através de nova simulação em MEF, onde a estrutura é subdividida novamente em frações menores. Como alternativa à novo refinamento de malha, para obtenção de grandezas em diferentes pontos, integrar RNA às soluções em MEF é uma alternativa interessante. Uma estrutura modelada através do MEF, discretizada em uma determinada quantidade de elementos, fornece soluções para número finito de pontos, a depender do tipo de elemento finito utilizado. Já uma rede treinada com esses mesmos pontos pode oferecer solução para infinitos pontos do domínio. Para classes de problemas em que se busca apenas a correspondência ponto-grandeza ainda não se discute a precisão dos resultados, mesmo onde a função de ativação é de maior complexidade em relação às aproximativas usadas no MEF.

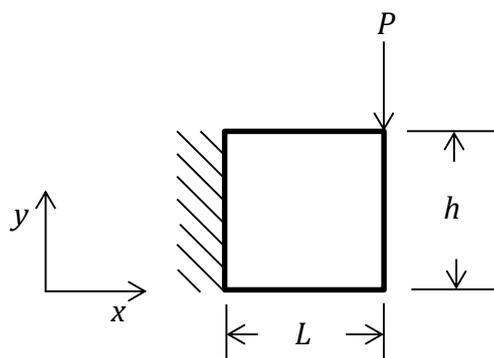
Do ponto de vista da precisão na solução em MEF o refinamento torna-se necessário para que o modelo apresente convergência nos resultados e modele adequadamente o fenômeno, garantindo assim, confiabilidade e robustez. É interessante frisar que tal convergência é observada através da relação entre o comportamento das grandezas de determinado ponto e o número de elementos MEF utilizados durante discretização. Entretanto, Galvão (2000) ressalta que a utilização de uma quantidade de elementos para geração de solução satisfatória dentro da precisão desejada reflete em aumento no custo computacional. Em situações onde o custo computacional relacionado ao refinamento torna-se relevante a RNA pode ser usada para estimar grandezas de determinado ponto em função do número de elementos. Espera-se assim, que os custos relacionados à precisão do método aproximativo possam ser diminuídos com a combinação dos dois métodos.

Neste tópico é discutido como integrar a RNA ao MEF linear, para garantir que pontos quaisquer do domínio sejam mapeados, e também como tentar obter soluções mais precisas a um custo computacional mais baixo.

5.3.1 Chapa sujeita a carga pontual

Uma chapa engastada de dimensões $4x4x0,15$, e coeficiente de Poisson $\nu = 0,25$ e módulo de elasticidade $E = 10000$ sob a ação de uma carga $P = 10$ aplicada em sua extremidade livre é demonstrada na Figura 5. 16. Trata-se de um exemplo hipotético preliminar, ilustrativo da metodologia aplicada a problemas simulados numericamente.

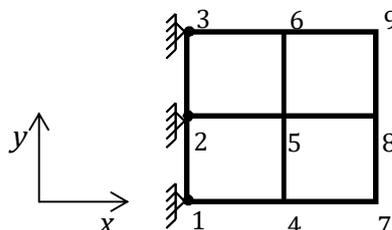
Figura 5. 16 – Modelo de chapa sob ação de carga pontual. Fonte: elaborada pela autora.



Admitiu-se que a estrutura está submetida ao regime de pequenas deformações e deslocamentos, comportamento linear elástico e estado plano de tensões. Portanto, para determinar os deslocamentos mediante a solicitação foi aplicado o Método dos Elementos Finitos, em versão linear.

A geometria da estrutura foi dividida em 4 elementos quadriláteros do tipo (Q4) com funções aproximativas lineares, como apresentado a seguir:

Figura 5. 17 – Divisão do domínio e numeração dos nós. Fonte: elaborada pela autora.



A simulação em MEF para a estrutura dividida em 4 elementos, com 9 nós, foi executada em 3,87 segundos. A Tabela 5. 3 expressa os valores de deslocamento obtidos.

Tabela 5. 3 – Simulação MEF.

Nó	Coordenadas	Deslocamento horizontal (m)	Deslocamento vertical (m)
1	(0,0)	0	0
2	(0,2)	0	0
3	(0,4)	0	0
4	(2,0)	-1,1994e-2	-1,5847e-2
5	(2,2)	-1,1500e-3	-1,4496e-2
6	(2,4)	1,3940e-2	-1,4429e-2
7	(4,0)	-1,5158e-2	-3,2743e-2
8	(4,2)	-1,2898e-3	-3,6266e-2
9	(4,4)	2,1264e-2	-4,9682e-2

Neste caso, como alternativa ao refinamento da malha e à interpolação utilizou-se Redes Neurais Artificiais para aproximar os deslocamentos mediante a solicitação. Assim esperava-se que as RNA generalizassem a solução de modo a obter o campo de deslocamentos para qualquer ponto na geometria da chapa.

Realizou-se os treinos com as redes tipo MLP. Como constatado na seção 5.2, as redes *Perceptron* não modelam adequadamente problemas não linearmente separáveis. Criou-se duas redes. As variáveis de entrada de ambas foram as coordenadas nodais na configuração indeformada. Para a rede 1 a saída foi o deslocamento na direção x e para a rede 2 o deslocamento na direção y . Para ambas, a configuração da RNA adotada foi a mesma. A solução final é o resultado de simulação das duas redes. Foi testada a possibilidade de modelar o problema com uma rede única, onde as entradas foram as coordenadas nodais e os respectivos deslocamentos nas direções x e y , mas os resultados apresentaram um resultado menos satisfatório do que a solução com duas redes.

Os treinamentos foram desenvolvidos utilizando redes com uma e duas camadas intermediárias, as funções de ativação usuais combinadas em todas as possibilidades e o número de neurônios incrementados de 1 a 10, e em mesmo número para o caso de duas camadas. Os padrões de treinamento são correspondentes aos 9 nós descritos na Figura 5. 17. Após vários treinamentos, a topologia que apresentou o melhor ajuste foi 2:4:1, com função de ativação linear na camada intermediária e tangente hiperbólica na saída. Na

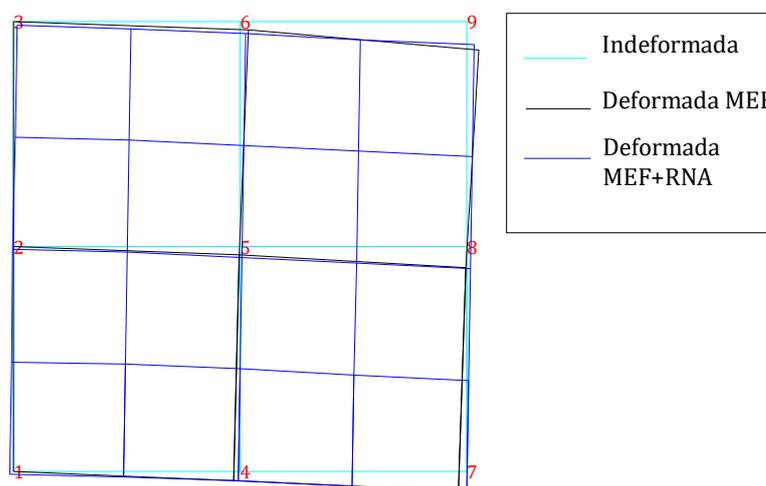
Tabela 5. 4 são expressas as coordenadas da configuração deformada, obtidas com uso da rede treinada, em comparação aos resultados obtidos via MEF.

Tabela 5. 4 – Resultados MEF e MEF+RNA.

Coordenadas configuração indeformada	Coordenada x – configuração deformada		Coordenada y – configuração deformada	
	MEF	MEF+RNA	MEF	MEF+RNA
(0,0)	0	-7,055e-3	0	-3,866e-3
(0,2)	0	-1,175e-3	2,000e+0	1,995e+0
(0,4)	0	5,812e-3	4,000e+0	3,994e+0
(2,0)	1,988e+0	1,996e+0	-1,584e-2	-1,589e-2
(2,2)	1,998e+0	2,002e+0	1,985e+0	1,981e+0
(2,4)	2,013e+0	2,009e+0	3,985e+0	3,979e+0
(4,0)	3,984e+0	3,999e+0	-3,274e-2	-3,596e-2
(4,2)	3,998e+0	4,006e+0	1,963e+0	1,962e+0
(4,4)	4,021e+0	4.012e+0	3,950e+0	3,960e+0

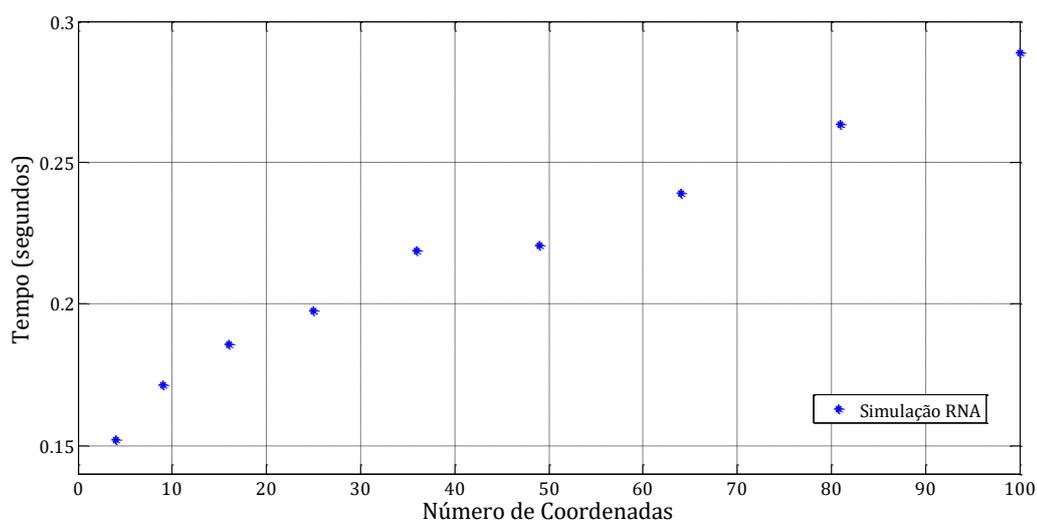
A Figura 5. 18 ilustra a configuração deformada obtida com a rede treinada, para estimar a posição de 25 pontos do domínio, além da configuração calculada com MEF.

Figura 5. 18 – Comparativo entre deformadas por MEF e MEF+RNA.



Os tempos alcançados para as redes treinadas para simulação de 4 a 100 nós variam de acordo com o gráfico da Figura 5. 19. Observa-se um aumento aproximadamente linear do custo computacional, neste caso. Vale destacar que apenas o tempo de simulação está sendo contabilizado, não incluindo o tempo gasto no treinamento da rede e tempo gasto para obtenção das amostras em simulações MEF.

Figura 5. 19 – Tempos de simulação RNA de 4 a 100 coordenadas.

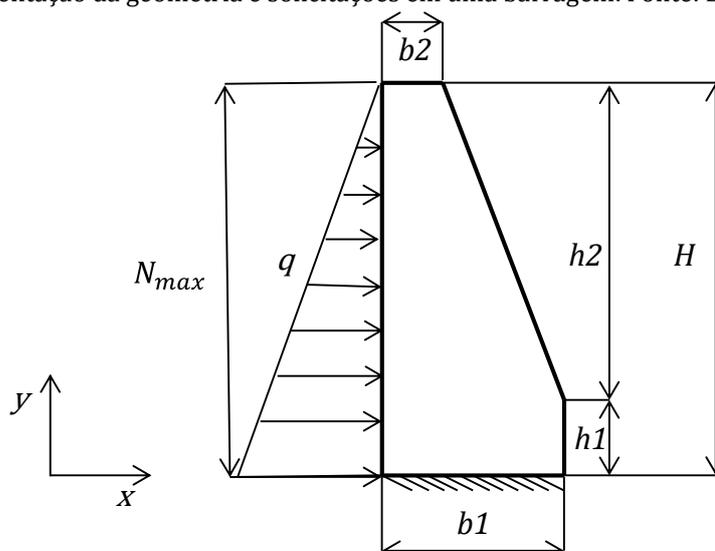


Com a RNA treinada, diversos pontos do domínio do problema podem ser simulados. Isso evita nova subdivisão da geometria, o que implica em menor ocupação de memória, devido ao não armazenamento de matrizes de rigidez maiores, e consequente diminuição nos tempos de obtenção da solução. Como os resultados de RNA treinada por dados MEF apresentaram coerência na resposta, justifica-se a aplicação de Redes Neurais como alternativa à nova simulação em MEF, uma vez que treinada, a RNA foi capaz de generalizar a solução e fornecer respostas para qualquer ponto do domínio do problema. Deve-se destacar que a simulação com RNA incorpora os erros do modelo em MEF utilizado para treinamento, de forma que é importante utilizar um boa amostra de dados para treinar a rede.

5.3.2 Modelo simplificado de barragem em contraforte

A Figura 5. 20 descreve o modelo simplificado de uma barragem do tipo contraforte, feita em concreto, com módulo de elasticidade $E = 2,1 \text{ GPa}$ e coeficiente de Poisson $\nu = 0,20$, submetida à pressão exercida pela água de peso específico $\gamma_{\text{agua}} = 9,81 \cdot 10^3 \text{ N/m}^3$, num determinado nível N_{max} , ao longo de uma de suas faces.

Figura 5. 20 - Representação da geometria e solicitações em uma barragem. Fonte: Elaborada pela autora.



A força q exercida pela água é regida pela relação:

$$q = (N_{max} - y) \cdot \gamma_{líquido} \quad (5.4)$$

O nível N_{max} representa a altura da coluna de líquido, sendo porcentagem da altura H da barragem, e $\gamma_{água}$ o peso específico do líquido. Para o problema, o líquido considerado foi a água a um nível de 100% da altura H .

A entrada dos dados referentes à geometria da estrutura que melhor descreve o fenômeno foi tomada em função de H , conforme (5.5):

$$b1 = 0,4H; \quad b2 = 0,2H; \quad h1 = 0,2H; \quad h2 = 0,8H; \quad (5.5)$$

Desprezando as forças de interação entre líquido e estrutura, e a não linearidade física do concreto, no regime de pequenas deformações e deslocamentos optou-se por desenvolver análises do campo de deslocamentos da estrutura através do MEF em sua versão linear em estado plano de tensões.

O domínio foi discretizado em elementos finitos quadrilaterais com quatro nós (Q4), sendo as funções aproximativas de primeira ordem, em deslocamentos. O MEF foi simulado em diferentes quantidades de elementos. De modo a manter coerência no tamanho dos elementos e adequá-los às características geométricas da estrutura, durante a subdivisão do domínio, estabeleceu-se uma relação entre o número de elementos que a compõe, nas direções x e y , de acordo com (5.6).

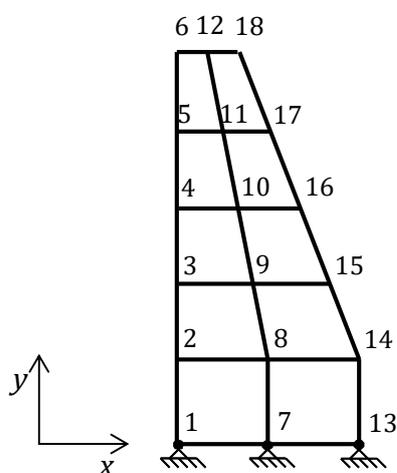
$$\text{numero}_{\text{elementos}} = nx \cdot ny \quad (5.6)$$

$$\begin{aligned} nx &= 2 \cdot k \\ ny &= 5 \cdot k, \quad k > 0 \in \mathbb{N}. \end{aligned}$$

onde nx e ny são divisões de elementos nas direções x e y , respectivamente.

A numeração dos nós e a distribuição dos elementos para domínio dividido em 10 elementos, por exemplo, de acordo com a relação proposta é representada na Figura 5. 21.

Figura 5. 21 – Numeração e distribuição dos nós em 10 elementos. Fonte: elaborada pela autora.



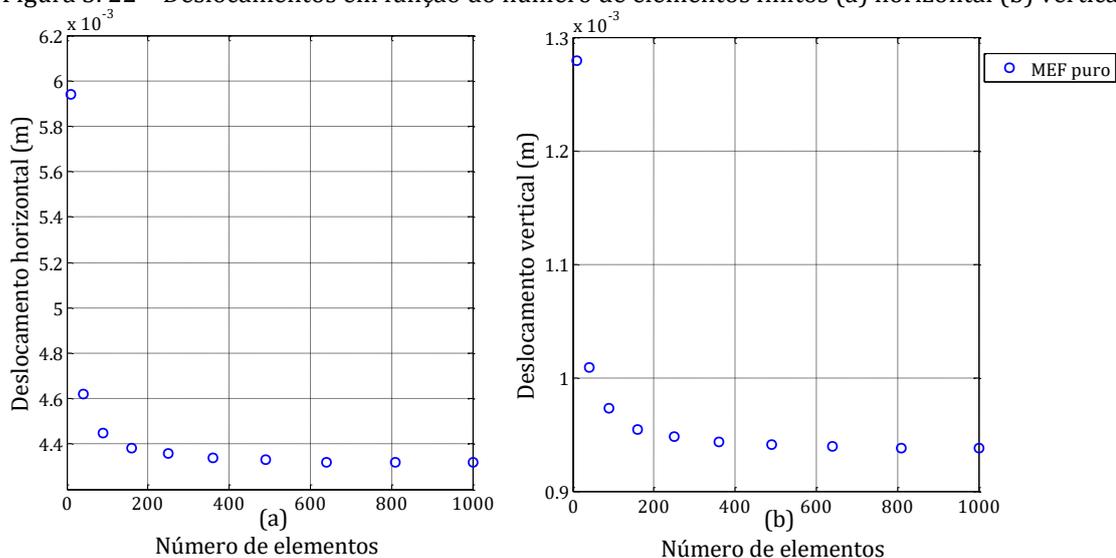
Para o nó da face superior, na superfície que sofre a solicitação, de coordenadas $(0,10)$, os deslocamentos horizontais e verticais em função da quantidade de elementos para $1 \leq k \leq 10$, são demonstrados na Tabela 5. 5.

Tabela 5. 5 – Número de elementos de discretização MEF e deslocamentos vertical e horizontal correspondentes.

Número de elementos	Deslocamento horizontal(m)	Deslocamento vertical (m)
10	5,940e-003	1,280e-003
40	4,620e-003	1,010e-003
90	4,450e-003	9,740e-004
160	4,380e-003	9,550e-004
250	4,360e-003	9,490e-004
360	4,340e-003	9,440e-004
490	4,330e-003	9,420e-004
640	4,320e-003	9,400e-004
810	4,320e-003	9,400e-004
1000	4,320e-003	9,400e-004

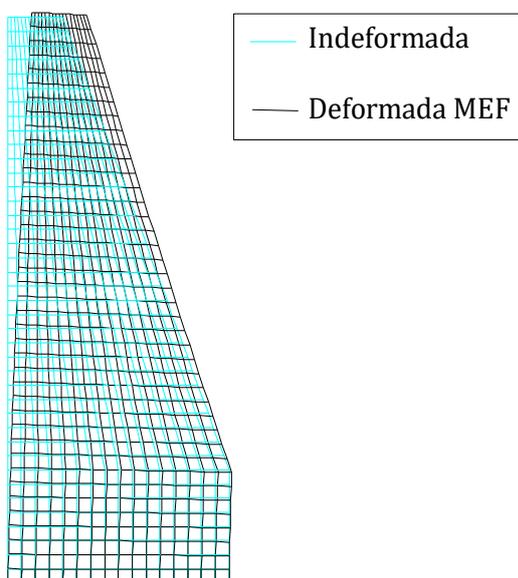
Graficamente, têm-se os resultados apresentados na Figura 5.22.

Figura 5. 22 – Deslocamentos em função do número de elementos finitos (a) horizontal (b) vertical.



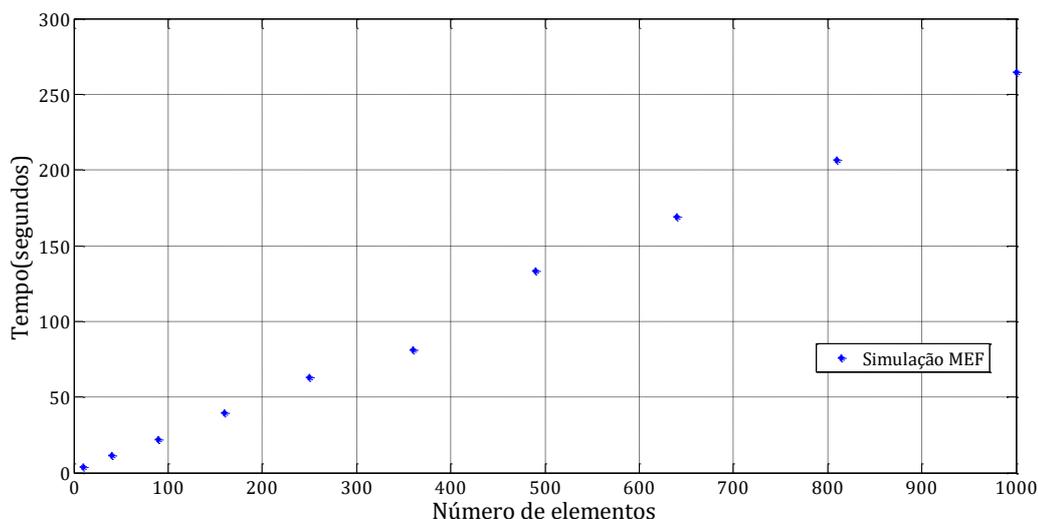
Os resultados expressos tanto na Tabela 5. 5 quanto na Figura 5. 22 evidenciam que o refinamento da malha a partir de 640 elementos finitos converge em seus valores para o ponto em análise. As configurações deformada e indeformada para MEF em 640 elementos é ilustrada na Figura 5. 23.

Figura 5. 23 – Representação das configurações indeformada e deformada da barragem.



Além disso, os tempos de simulação correspondentes aos refinamentos adotados indicam crescimento aproximadamente linear da curva tempo *versus* número de elementos, conforme Figura 5. 24.

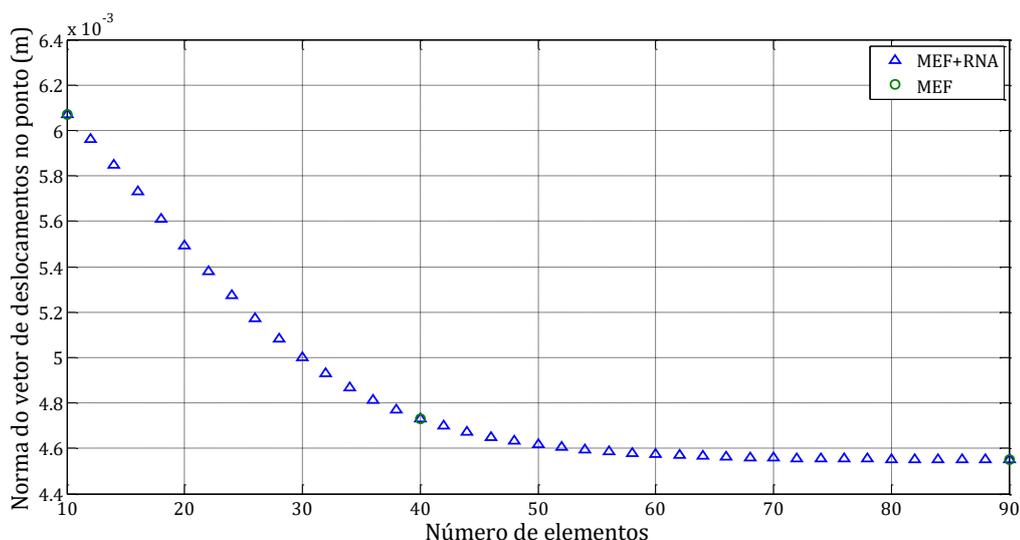
Figura 5. 24 – Tempos de simulação em MEF para os refinamentos entre 10 e 1000 elementos.



A relação entre uma grandeza física e a quantidade de elementos é um dos importantes indicadores de desempenho das formulações do Método dos Elementos Finitos. A partir do comportamento desta relação pode-se inferir acerca da viabilidade de determinado tipo de elemento e dos custos para obtenção da solução. Como dito na introdução do item 5.3, o refinamento de malha ou uso de funções aproximativas de maior grau, por exemplo, implicam em resultados mais precisos, entretanto em maior ocupação de memória e tempo de processamento. Uma sugestão para obter melhores resultados através da integração de RNA ao MEF por extrapolação é aqui aplicada e discutida.

A quantidade de amostras de simulação em MEF para treinamento da RNA deve ser a mínima possível para viabilizar as aplicações MEF+RNA. Com objetivo de aumentar a quantidade de amostras usadas para a extrapolação uma rede de topologia 1:1:1 e função de ativação logística na camada intermediária foi treinada para interpolar valores da norma do deslocamento no ponto em função do número de elementos. A menor quantidade para a qual a rede proposta apresentou desempenho satisfatório foi de 3 amostras, de simulações com a estrutura discretizada em 10, 40 e 90 elementos, como exibido na Figura 5.25.

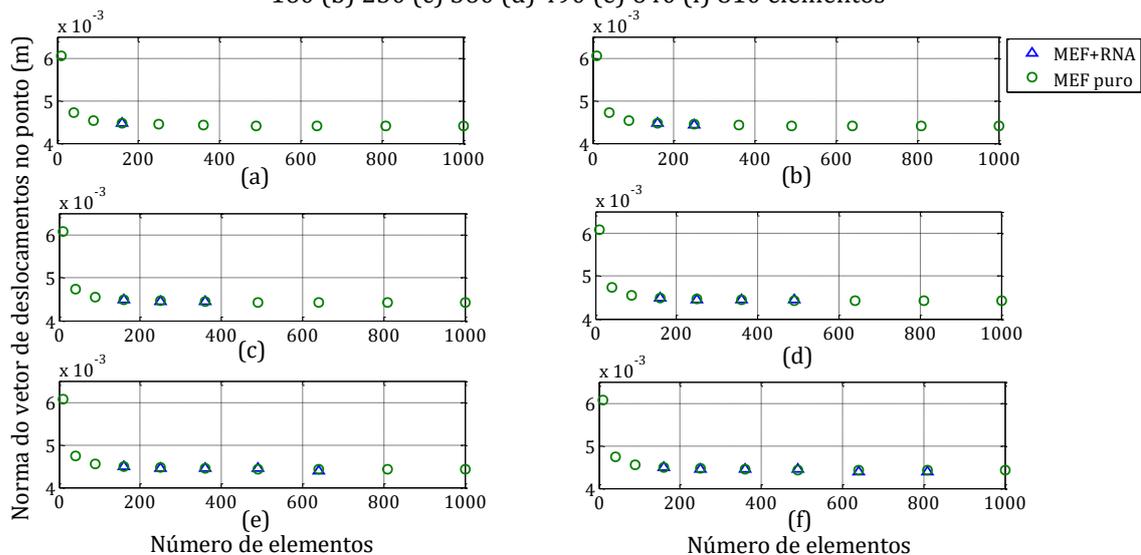
Figura 5. 25 – Interpolação da norma do vetor de deslocamentos no ponto entre 10 e 90 elementos.



Foram geradas 41 amostras MEF+RNA a partir de 3 simulações em MEF.

Foram utilizadas 6 redes, uma para cada valor subsequente previsto, de topologia 1:3:3:1, com funções de ativação logística e tangente hiperbólica nas camadas intermediárias. O número de elementos é fornecido como variável de entrada e os deslocamentos correspondentes como variável de saída, para as quantidades de elementos 250, 360, 490 e 640 e 810. Os resultados obtidos por extrapolação são ilustrados na Figura 5. 26.

Figura 5. 26 – Norma dos deslocamentos previstos no ponto em função da quantidade de elementos (a) 160 (b) 250 (c) 360 (d) 490 (e) 640 (f) 810 elementos



Os resultados MEF e MEF+RNA são exibidos na Tabela 5. 6.

Tabela 5. 6 – Comparativo entre a norma do vetor de deslocamentos obtida via MEF e MEF+RNA no ponto

Número de elementos	MEF	MEF+RNA
160	4,4800e-003	4,4848e-003
250	4,4600e-003	4,4399e-003
360	4,4400e-003	4,4400e-003
490	4,4300e-003	4,4434e-003
640	4,4300e-003	4,4052e-003
810	4,4200e-003	4,3986e-003

Os gráficos da Figura 5. 26 evidenciam tendência a convergência entre a solução MEF e MEF+RNA. A oscilação advinda de valores que se distanciam da fronteira do conjunto inicial de treinamento é explicada pela forte correlação entre a quantidade de amostras para treinamento e a capacidade de generalização da rede. Entretanto, ainda assim pode-se tornar viável a aplicação de Redes Neurais para modelagem por extrapolação nos casos em que o refinamento da resposta via MEF demande aumento significativo, ou até impeditivo, do custo computacional. Para casos onde o erro provocado pelas oscilações de MEF+RNA são menores em relação ao erro MEF em malhas pobres, modelar através de extrapolação por MEF+RNA é uma boa alternativa para obter respostas mais precisas.

Além disso, os tempos de treinamento e simulação da RNA foram da ordem de milissegundos. Mesmo com a necessidade de uma rede para cada valor subsequente estimado, a solução com MEF+RNA pode ser viável como alternativa à nova simulação MEF, mostrando-se competitiva.

Conclui-se que é possível prever valores mais precisos em tempo hábil, correspondentes a uma configuração mais refinada do MEF, sem a necessidade de nova simulação do método. Pode-se dizer, inclusive, que existem casos em que uma RNA treinada a partir de resultados com erros consideráveis pode fornecer resultados com erros menores em relação a um valor de referência.

5.4 Problemas com Solução Aproximada via Formulação Não Linear do MEF

Este tópico expõe exemplos onde a modelagem estrutural através do método dos elementos finitos não linear geométrico (MEF NLG) revela-se mais adequada às características do problema.

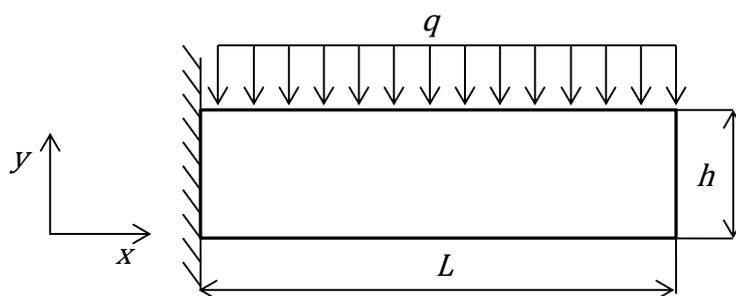
O MEF não linear geométrico desconsidera as simplificações impostas ao método linear. Torna-se portanto, um método mais robusto, por abranger análises tanto em regime de pequenas como grandes deformações e deslocamentos.

Para os exemplos abordados foi aplicado o método dos elementos finitos não linear direto e incremental, brevemente descrito na seção 3.5.2 .

5.4.1 Viga em balanço sob carregamento distribuído

Uma viga de seção quadrada, com altura $h = 0,1$ m e largura $e = 0,1$ m, construída em aço, de coeficiente de Poisson $\nu = 0,33$ e módulo de elasticidade $E = 210$ GPa, está submetida a um carregamento distribuído $q = 10$ kN/m constante ao longo do todo o comprimento L , como mostrado na Figura 5. 27.

Figura 5. 27 – Modelo bidimensional de viga em balanço sujeita a carregamento distribuído. Fonte: elaborada pela autora.



Admita-se que o comprimento da viga é considerado variável neste problema, sendo definido por um fator multiplicador λ , proporcional à altura da viga h :

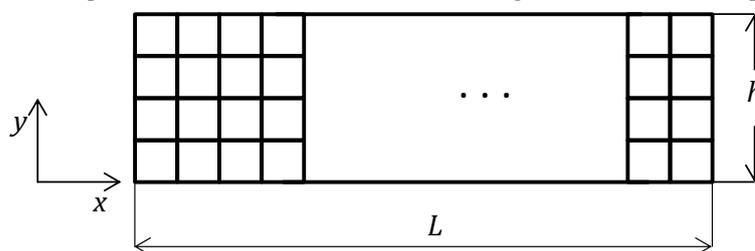
$$L = \lambda \cdot h \quad (5.5)$$

As dimensões da viga podem ser relacionadas através deste parâmetro λ , de modo que quando $h \ll L$ a estrutura pode ser caracterizada como geometricamente esbelta.

Esta característica confere uma variação não linear no campo de deslocamentos da estrutura, provocado quando há grandes deslocamentos, mas pequenas deformações, como em geral ocorrem com as estruturas esbeltas. Nestes casos, a análise não linear geométrica torna-se mais adequada à modelagem do problema.

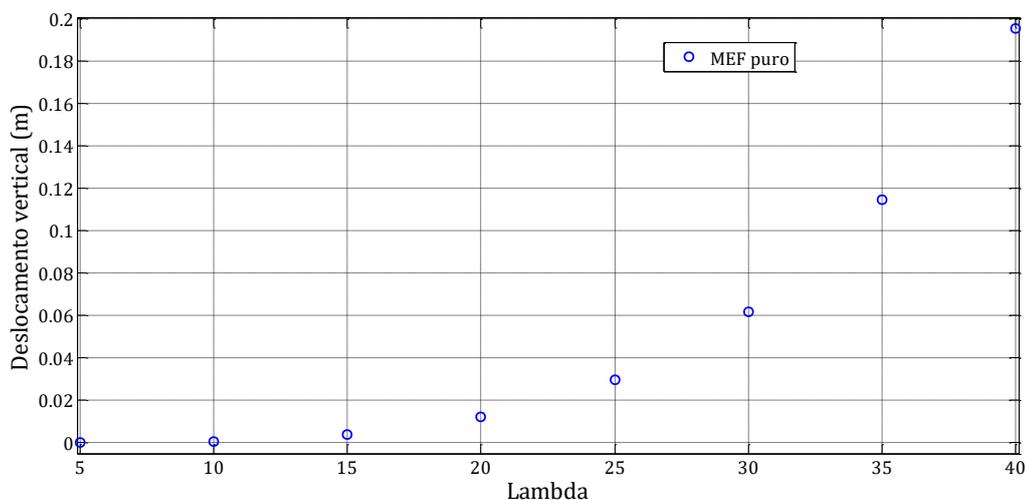
A geometria da estrutura foi dividida em elementos quadrilaterais de nove nós (Q9) de mesmo tamanho, quantidade fixa em 4 na direção y e quantidade variável de acordo com o comprimento L ao longo da direção x , como ilustrado na Figura 5. 28

Figura 5. 28 – Esquema de divisão dos elementos da viga. Fonte: elaborada pela autora.



Adotada a hipótese de que o carregamento mantém suas características de aplicação em todos os nós da face superior para qualquer comprimento L , para carregamento incremental de 0 a q em 50 incrementos de carga, obtiveram-se os seguintes deslocamentos verticais em função de λ para o nó de coordenada (L, h) como mostrado na Figura 5. 29.

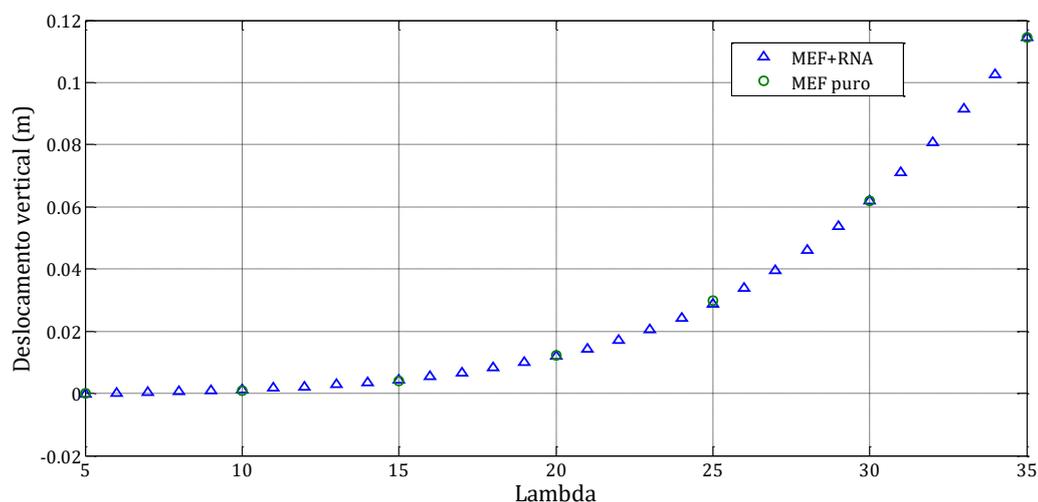
Figura 5. 29 – Deslocamentos verticais em função de λ para a coordenada (L, h) .



Para a modelagem deste problema por Redes Neurais para fins de interpolação é proposto uma rede de múltiplas camadas 1:1:1:1 com funções de ativação linear e

tangente hiperbólica nas camadas intermediárias, λ como variável de entrada e o deslocamento após a ação do carregamento como variável de saída, treinada com 7 pares de entrada-saída. Para entradas compreendidas entre $5 \leq \lambda \leq 35$ a rede proposta em fase de operação forneceu os seguintes resultados:

Figura 5. 30 – Deslocamentos verticais em função do comprimento da viga.



A curva da Figura 5. 30 descreve o comportamento dos deslocamentos verticais no ponto (L, h) à medida que a estrutura sofre alteração em suas dimensões para possibilidades de λ que não estavam contidas no conjunto de padrões obtidos em MEF puro. Valores de λ usados para o treinamento também foram incorporados como entrada para simulação. Para esses pontos a estimativa de deslocamento obtida via Redes Neurais combinadas ao MEF são descritos Tabela 5. 7.

Tabela 5. 7 – Valores por simulação em MEF não linear geométrico e RNA.

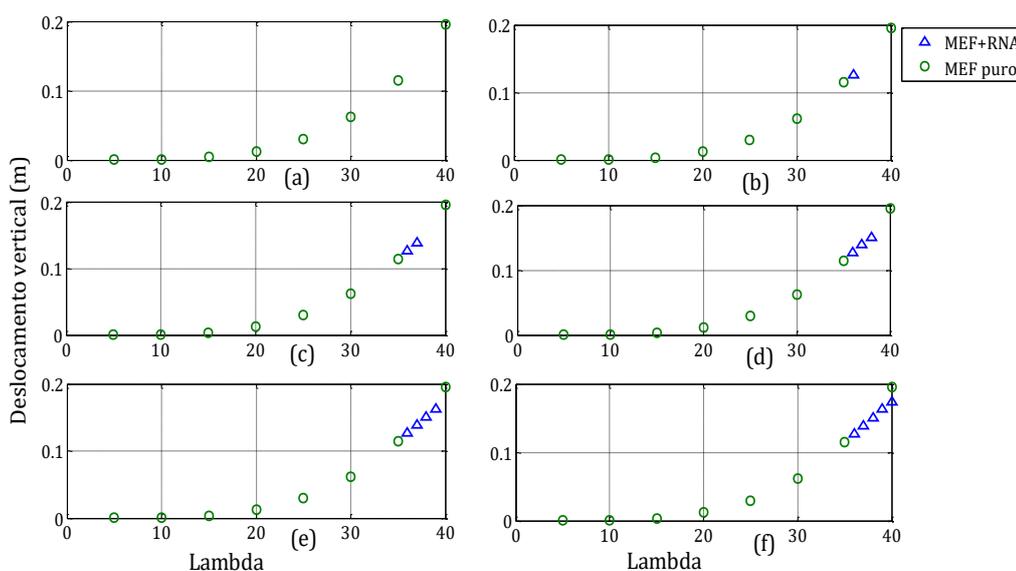
λ	MEF NLG (m)	MEF NLG+RNA (m)
5	4,940e-005	-1,725e-004
10	7,688e-004	1,162e-003
15	3,872e-003	4,371e-003
20	1,222e-002	1,190e-002
25	2,981e-002	2,866e-002
30	6,179e-002	6,181e-002
35	1,144e-001	1,144e-001

Para valores de λ superiores a 35, simulou-se a rede já criada durante interpolação com um valor de λ contido na vizinhança de 35. Este novo valor de deslocamento estimado

por simulação é agregado ao conjunto inicial de treinamento juntamente com os resultados advindos por interpolação e os pesos reajustados através de novo treinamento, caracterizando a extrapolação. Sucessivamente, novos valores de deslocamento correspondentes puderam ser estimados, obedecendo à condição de estarem na fronteira de $\lambda_{máximo}$ do conjunto de treinamento usado.

Nesta fase uma RNA de múltiplas camadas foi configurada com um neurônio de entrada (λ) e um neurônio de saída (deslocamentos verticais), 2 camadas intermediárias com 2 neurônios cada e função de ativação linear e tangente hiperbólica, respectivamente, sendo treinada e posteriormente simulada com um valor de λ de fronteira. Assim, cada par estimado de λ -deslocamento exige uma rede. Os deslocamentos estimados para $36 \leq \lambda \leq 40$, por 5 redes treinadas para $36 \leq \lambda \leq 40$ são ilustrados na Figura 5. 31.

Figura 5. 31 – Estimativa de valores de deslocamentos por extrapolação usando RNA+MEF. (a) curva de deslocamento puro MEF, (b) previsão de deslocamento para $\lambda = 36$, (c) previsão de deslocamento para $\lambda = 37$, (d) previsão de deslocamento para $\lambda = 38$, (e) previsão de deslocamento para $\lambda = 39$, (f) previsão de deslocamento para $\lambda = 40$.



A solução na Figura 5. 31 aponta uma tendência inicial de convergência na curva RNA+MEF em relação às simulações em puro MEF. Entretanto, observa-se que à medida que o valor previsto se afasta do conjunto inicial de treinamento há divergência com os deslocamentos em puro MEF. Isso ocorre porque o desempenho de uma RNA é fortemente condicionado aos dados empregados durante o treinamento e a rede representa o fenômeno adequadamente quando é simulada com entradas cujos valores estão dentro de seus valores de máximo e mínimo. Mesmo com a restrição de proximidade da fronteira entre dados de simulação e o domínio dos dados de treinamento a solução apresentou uma ligeira variação na curvatura do gráfico no intervalo em análise. O erro relativo entre

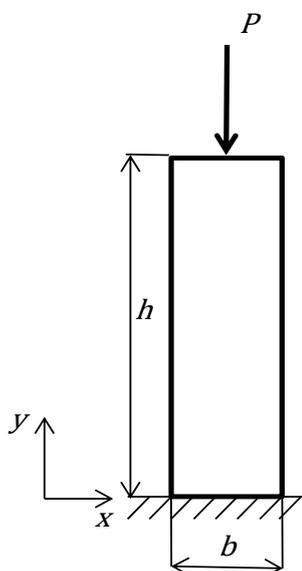
as soluções para $\lambda = 40$ foi de 10,8%. Além disso, como visto nos itens 5.2.1 e 5.2.2 a quantidade de amostras para treinamento afeta diretamente a qualidade da solução. Portanto, a inserção de amostras MEF+RNA no conjunto de treinamento MEF puro implica em erros maiores quando os valores de entrada para simulação se afastam da fronteira, o que é indicativo de propagação do erro ao longo dos resultados previstos. Espera-se que, se o valor subsequente simulado e incorporado ao próximo treinamento for tomado cada vez mais próximo da fronteira, os resultados se apresentem melhores.

Ainda com este desvio em relação ao resultado de referência, pode ser interessante considerar a aplicação de Redes Neurais para extrapolação. As funções aproximativas possuem maior grau de complexidade e podem convergir para os resultados mais precisos se comparados a funções aproximativas lineares, comumente usadas em MEF. Os algoritmos de Redes Neurais também são mais simples de serem desenvolvidos, e uma vez treinada, a rede obtém o mapeamento entre entradas e saídas de maneira mais veloz se comparado a uma nova simulação em elementos finitos.

5.4.2 Coluna com carga excêntrica

A estrutura de seção transversal quadrada $0,1 \times 0,1$ m e comprimento 2 m, é admitida em concreto, com módulo de elasticidade E de 25 GPa, poisson $\nu = 0,2$, e está sujeita a uma carga concentrada P , aplicada de forma centrada, como esboçado na Figura 5.32.

Figura 5.32 – Pilar sujeito a carga concentrada. Fonte: Elaborada pela autora.



As dimensões dadas classificam a estrutura como esbelta, $h \gg b$. Para estruturas esbeltas admite-se que a aplicação de uma carga de compressão como a descrita pode ultrapassar os máximos valores admissíveis e provocar instabilidade. Nestas condições a estrutura sofre deslocamentos horizontais, caracterizando o fenômeno denominado flambagem.

Em seus estudos Leonhard Euler estabeleceu o limite no valor da carga aplicada para o qual a estrutura atinge limite de estabilidade e sofre flambagem. A teoria de Euler define esta carga como carga crítica P_{cr} e a descreve como:

$$P_{cr} = \frac{\pi^2 EI}{L^2} \quad (5.6)$$

Onde L é o comprimento efetivo da coluna e está relacionado às condições de vinculação. Neste caso, de coluna engastada -livre, o comprimento efetivo vale o dobro do comprimento real h .

I é o momento de inércia referente à direção na qual a estrutura pode sofrer instabilidade, e E o módulo de elasticidade do material.

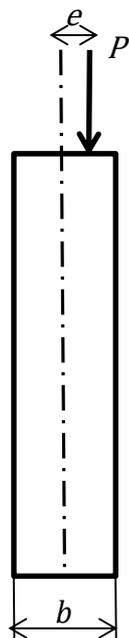
Para a coluna em análise, admitiu-se que a carga, inicialmente centrada, pode ser aplicada fora do eixo da coluna, com uma excentricidade e definida por:

$$e = \alpha \cdot b \quad (5.7)$$

α é um fator de excentricidade associado à largura da seção transversal da coluna. Diferentes valores de α induzem diferentes valores de deslocamento horizontal no topo da coluna.

A Figura 5. 33 ilustra o deslocamento do ponto de aplicação de P_{cr} ao longo da base b da estrutura.

Figura 5.33 – Excentricidade no ponto de aplicação da carga concentrada. Fonte: elaborada pela autora.

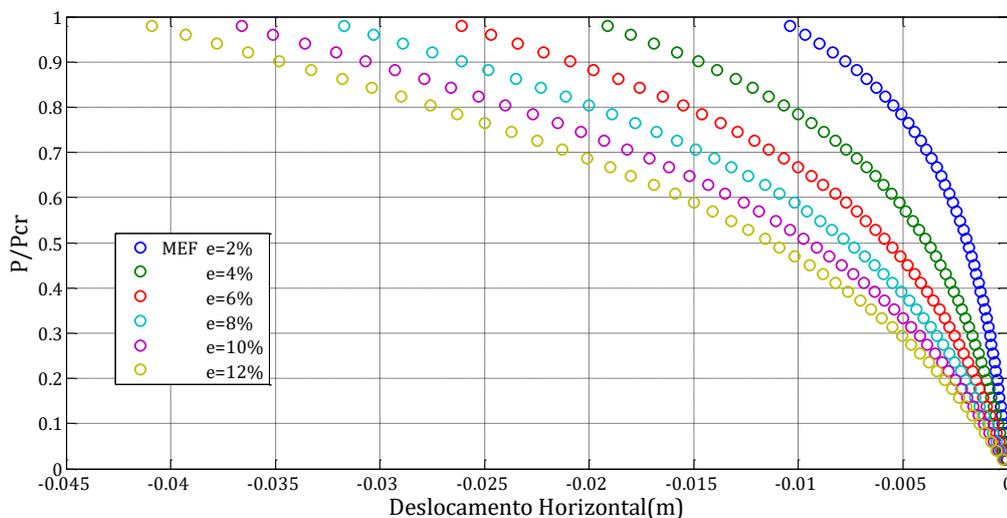


O método dos elementos finitos não linear geométrico foi empregado para modelar o comportamento dos deslocamentos horizontais deste pilar mediante ao carregamento quando a carga é aplicada de modo incremental de 0 até 98% de seu valor crítico de referência. Vale destacar que este valor crítico corresponde à condição de carga centrada, e que a intenção do exemplo é medir a resposta em deslocamento horizontal da coluna para diferentes valores de excentricidade. Não se discute trajetória de equilíbrio da estrutura sob as diferentes condições de carregamento.

A geometria foi dividida em elementos quadriláteros de 9 nós (Q9) de mesmo tamanho, 4 na direção x e 80 na direção y , e funções aproximativas quadráticas. A Figura 5.34 reproduz os resultados para simulações por elementos finitos com não linearidade para 0 a 98% da carga crítica em 50 incrementos, para excentricidades 2, 4, 6, 8, 10 e 12% da largura.

Com base na Figura 5.34 pode-se relacionar os deslocamentos à carga e ao local de aplicação desta carga. Quanto mais o ponto de aplicação se desvia do eixo central da base e quanto mais a carga se aproxima do valor crítico de referência, maiores são os deslocamentos horizontais produzidos.

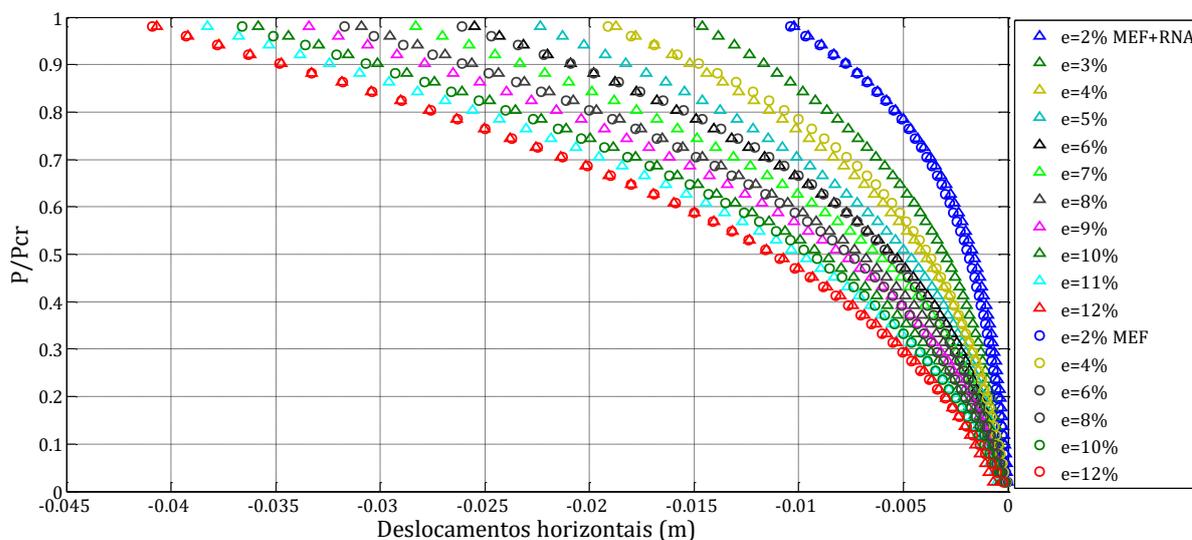
Figura 5.34 – Deslocamentos horizontais para aplicação incremental de carga até 98% da carga crítica para excentricidades 2, 4, 6, 8, 10 e 12% da largura da seção.



A modelagem pela rede neural múltiplas camadas adotada possui os seguintes parâmetros: 2 neurônios de entrada, sendo eles a porcentagem de carga em relação a carga crítica e o fator de excentricidade de carga, 1 neurônio na saída referente ao deslocamento correspondente às entradas, 2 camadas intermediárias de 4 neurônios cada, com função de ativação linear e tangente hiperbólica, respectivamente. A função de ativação da saída foi mantida linear. Os padrões para treinamento correspondem às excentricidades de 2, 6 e 12% da largura da seção transversal do pilar.

A RNA simulada para excentricidades de 2 a 12% é exibida na Figura 5.35.

Figura 5.35 – Deslocamentos obtidos por MEF+RNA para excentricidades variando de 2 a 12%



O tempo gasto durante a simulação RNA foi de 0,46 segundos.

Os resultados obtidos pela junção de Redes Neurais ao MEF apresentaram-se satisfatórios de acordo com a Figura 5. 35. A modelagem por Redes Neurais mostrou-se vantajosa, uma vez que, a partir de dados de apenas 3 simulações em MEF (2, 6 e 12%) para geração de padrões de treinamento da rede obteve-se soluções para excentricidades no local de aplicação de carga entre 2 e 12% a tempos de simulação significativamente menores se comparados a nova simulação em MEF.

6 CONSIDERAÇÕES FINAIS

Neste trabalho pretendeu-se obter soluções integradas entre o Método dos elementos Finitos e Redes Neurais Artificiais para efetuar análises em engenharia estrutural.

Para maior embasamento durante o estudo foram levantados os principais conceitos referentes a Redes Neurais Artificiais, com histórico, definições, características de um neurônio artificial, tipos de aprendizado e os principais modelos, além da aplicabilidade das RNA nas análises estruturais.

Além de fundamentação teórica foram realizados procedimentos práticos para determinar características e comportamento das respostas fornecidas por RNA mediante à diferentes configurações de treinamento. Através de problemas de complexidade conhecida foram exploradas duas principais arquiteturas de rede disponíveis: as redes de camada única (*Perceptron*) e as múltiplas camadas (MLP). Também foram investigadas as influências dos parâmetros, como número de neurônios e camadas, quantidade de padrões de treinamento, entre outros, na qualidade e desempenho da solução fornecida via Redes Neurais.

As aplicações mais complexas abordadas foram problemas de elasticidade plana, através de códigos originais de Método dos Elementos Finitos desenvolvidos ao longo do Curso de mestrado, escritos em 2013.

De modo geral, as RNA foram incorporadas ao MEF com dois objetivos:

- Interpolação: os valores de teste e simulação contidos dentro do intervalo de máximo e mínimo do conjunto de treinamento; e
- Extrapolação: valores de teste e simulação não contidos dentro do intervalo de máximo e mínimo, como tentativa de previsão através de um incremento nos valores de entrada iniciais.

O software MATLAB® mostrou-se uma ferramenta adequada para o estudo, pois permitiu, de maneira simplificada, a implementação computacional do Método dos

Elementos Finitos e treinamento e operação de Redes Neurais num mesmo ambiente.

Os resultados alcançados em ambas as abordagens (interpolação ou extrapolação) apontaram convergência das respostas entre MEF+RNA e MEF puro. A resposta por interpolação apresentou estabilidade e erros significativamente pequenos, o que possibilitou mapear com boa precisão as entradas e saídas durante o processo de operação da RNA, nos exemplos testados. Além disso, os tempos de simulação da rede foram extremamente pequenos, na ordem de milisegundos, tornando a abordagem interessante. Para casos onde se empregou extrapolação, o traçado da curva evidenciou um tendenciamento da resposta em relação ao comportamento dos dados de treinamento. Entretanto, esta instabilidade da solução é devida a características deste conjunto de treinamento e ao paradigma de rede utilizado. Para o problema de previsão, a literatura indica redes dinâmicas. A proposta de previsão por meio de redes neurais MLP, da maneira que é descrita na metodologia, indicou a tendência da solução, mas a viabilidade de aplicação deve ser avaliada pelo usuário. Se a prioridade da solução for tempo de resposta o uso da extrapolação é fortemente indicado como substituição à novas simulações em elementos finitos. No entanto, se a prioridade é precisão deve-se ponderar e julgar minuciosamente as características do problema em análise.

Uma das maiores desvantagens em usar uma rede neural treinada com dados MEF está no custo de obtenção das amostras de treinamento. Determinadas simulações em MEF dispendem tempo de simulação e grande taxa de ocupação de memória das máquinas. No entanto, sabe-se que a rede já treinada é capaz de mapear com precisão as variáveis do modelo. Uma vez ajustados, os pesos carregam consigo a memória do fenômeno e a rede em processo de operação garante rapidez e precisão de resultados. Logo, a desvantagem do processo de obtenção dos dados pode ser compensada pela rapidez com que a solução pode ser apresentada para quaisquer valores de entrada, evitando assim, novas simulações em MEF.

No intuito de se aprimorar a técnica são propostas para trabalhos futuros:

- Treinar redes mais generalistas, com as variáveis de entrada e saída parametrizadas em relação a valores de referência;
- Tornar as análises mais completas com a inclusão de não linearidade física dos materiais;

- Análises de comportamentos dinâmicos das estruturas através de arquiteturas de redes mais complexas;
- Aplicação da metodologia utilizada para análise de problemas de maior complexidade/tempo de processamento, a fim de se verificar as vantagens significativas na combinação entre MEF e RNA;
- Análise detalhada e comparativa nos tempos de processamento entre a utilização de MEF puro e MEF+RNA; e
- Criação de interface gráfica interativa que permita a inserção de dados e visualização de resultados pelo usuário final, dos modelos mecânicos em análise MEF+RNA.

REFERÊNCIAS BIBLIOGRÁFICAS

Abdallah, K. M.; Stavroulakis, G. E. A Backpropagation neural network model for semi-rigid steel connections. *Microcomputers in Civil Engineering*. Vol. 10, No. 2, p. 77-87, 1995.

Almeida, M. L. **Elementos finitos paramétricos implementados em Java**. Dissertação (Mestrado) - Universidade Federal de Minas Gerais. Belo Horizonte, 2005.

Almeida, R. L. F. **Aplicação de Redes Neurais Artificiais para Aproximar o Comportamento de Estruturas**. Dissertação (Mestrado) - Faculdade Ciencia e Tecnologia, Universidade Nova de Lisboa, 2012.

Azevedo, A. F. M. **Método dos elementos finitos**. 1ª ed., 2003.

Bağcı M.; Altıntaş G. Artificial Neural Network Analysis of Infilled Planar Frames, *Proceedings Of ICE, Structures e Buildings*, 159(1), p. 37-44, 2006

Bakhary, N.; Hao, H.; Deeks A. Damage detection using artificial neural network with consideration of uncertainties. **Engineering Structures**, Vol. 29, Nº. 11, p. 2806–2815, 2007.

Barbosa, A. H. **Análise de Confiabilidade Estrutural Etilizando o Método de Monte Carlo e Redes Neurais**. Dissertação (Mestrado) - Programa de Pós-Graduação em Engenharia Civil, UFOP. Ouro Preto, 2004.

Barreto, J. M.; **Introdução às Redes Neurais Artificiais**. Laboratório de Conexão e Ciências Cognitivas - UFSC -Departamento de Informática e de Estatística - Florianópolis – SC, 2002.

Begum, R; Chamberlain, B.; Hirson, A. Integrity Testing of Concrete Surfaces Using Artificial Neural Networks. **CIVIL-COMP95** Developments in Neural Networks and Evolutionary Computing for Civil and Structural Engineering, Topping, B. H. V. (ed.), CIVIL-COMP Press, Edinburgh, UK, p. 1-6 , 1995.

Braga, A. P.,Carvalho, A. P. de L., Ludemir, T. B. **Fundamentos de Redes Neurais Artificiais**. 11º Escola de Computação - Rio de Janeiro. DCC/IM, COOPE/Sistemas,

NCE/UFRJ, 1999.

Castro, L. L. B. **Inteligência Computacional Aplicada à Engenharia de Estruturas**. Tese (Doutorado) - Universidade de Brasília. Departamento de Engenharia Civil e Ambiental. Brasília. 2009.

Cavalcante, A. A. Márcio. Notas de aula da disciplina “Análise não linear das estruturas”. PPGEC/UFAL-Maceio – 2013.

Chen, H. M., Tsai, K. H., Qi, G. Z., Yang, J. C. S. and Amini, F. Neural Network for Structure Control. **Journal of Computing in Civil Engineering**, Vol. 9, No. 2, April, 168-176, 1995.

Cybenko, G. **Continuous Valued Neural Networks with Two Hidden Layers are Sufficient**. Technical Report, Department of Computer Science, Boston: Tufts University, 1988.

Fellipa, C. A. **Introduction to The Finite Element Method**. Lecture notes. University of Colorado, 2000.

Flood, I.; Kartam, N. Neural networks in civil engineering: Systems and application. **J. Comp. in Civ. Engrg.**, ASCE, 8(2), 131-148, 1994.

Funahashi, K. On the approximate realization of continuous mapping by neural network. **In neural network**, Vol 2, p 183-189, 1989.

Galvão, A. S. **Formulações geometricamente não lineares de elementos finitos para análise de sistemas estruturais metálicos reticulados planos**. Dissertação (Mestrado), Ouro Preto: Programa de Pós-Graduação em Engenharia Civil, DECIV/Escola de Minas/UFOP, 2000.

Ghaboussi, J., Garrett, J. H. and Wu, X., Knowledge-based modeling of material behavior with neural networks. **J. of Engineering Mechanics**, Vol. 117, No. 1, January, 132-152, 1991.

Goh, A. T. C.; Wong, K. S.; Broms, B. B. Multivariate Modelling of FEM Data Using Neural Networks. **CIVIL-COMP95 Developments in Neural Networks and Evolutionary Computing for Civil and Structural Engineering**, Topping, B. H. V. (ed.), CIVIL-COMP Press, Edinburgh, UK, 59-64, 1995.

Gomes, H. M. **Técnicas de Avaliação da Confiabilidade em Estruturas de Concreto Armado**. Tese (Doutorado), Universidade Federal do Rio Grande do Sul. Programa de Pós-Graduação em Engenharia Civil. Porto Alegre. 2001.

Haykin, S. **Redes Neurais - Princípios e Prática**. 2. ed, editora Bookman, 1999.

Hebb, D. O. The organization of behavior. New York:Wiley, 1949.

Hecht-Nielsen, R. Theory of the backpropagation neural network. **In Proceedings of the International Joint Conference on Neural Network.**, vol. 3, p. 11-13, 1989.

Hopfield, J. J. **Neural Networks and Physical Systems with Emergent Collective Computational Abilities**. Proceedings of the National Academy of Sciences of the USA, Vol. 79 no. 8, p. 2554–2558. 1982.

Khan, A. I.; Topping, B. H. V.; Bahreininejad, A. Parallel Training of Neural Networks for Finite Element Mesh Generation. CIVIL-COMP93, The Third International Conference in the Application of Artificial Intelligence to Civil and Structural Engineering, 17th-19th August, Topping, B. H. and Khan, A. I. (eds), Heriot-Watt University, Edinburgh, UK, 81-94 , 1993.

Kirkegaard, P. H.; Rytter, A. The Use of Neural Networks for Damage Detection and Location in a Steel Member. CIVIL-COMP93, The Third International Conference in the Application of Artificial Intelligence to Civil and Structural Engineering, 17th-19th August, B. H. Topping e A. I. Khan (eds), Heriot-Watt University, Edinburgh, UK, p. 1-9 ,1993.

Lastiti, C. P.; Pauletti, R. M. O. Aplicação de redes neuronais artificiais à engenharia de estruturas. 2004.

Lefik, M. Use of artificial neural network to define a nonlinear effective constitutive law for a composite, Proc. of XIII Pol. **Conf. on Computer Methods in Mechanics** 1, p. 723–127, 1997.

Lefik, M. Some aspects of application of artificial neural network for numerical modeling in civil engineering. **Bulletin of the Polish Academy of Sciences: Technical Sciences**. Vol. 61, Issue 1, p. 39–50, 2013.

Lima Jr., E. T. **Formulação do Método dos Elementos de Contorno para Análise de Cascas Abatidas**. Dissertação (Mestrado) – Escola de Engenharia de São Carlos,

Universidade de São Paulo, 2006.

Lins, P. G. C. **Classificação de maciços rochosos: Uma abordagem por redes neurais**. 167p. Dissertação (Mestrado) - Escola de Engenharia de São Carlos da Universidade de São Paulo, 2002

Lopes, P. S. **Detecção de danos em estruturas por meio de técnicas de redes neurais artificiais e algoritmos genéticos**. Dissertação (Mestrado), Universidade Federal de Itajubá – Instituto de Engenharia Mecânica, 2007.

McCulloch, W. S.; Pitts, W. H. A Logical Calculus of the Ideas Immanent in Nervous Activity. *Bulletin of Mathematical Biophysics*, 5, pp. 115-133, 1943.

Martins, J. A. **Avaliação de diferentes técnicas para reconhecimento de fala**. Tese (Doutorado), Faculdade de Engenharia Elétrica e Computação da Universidade Estadual de Campinas – UNICAMP, 1997.

Medeiros, L. F. de. **Redes neurais em delphi**. Florianópolis: Visual Books Editora, 115p, 2003.

Minsky, M.; Papert, S. **Perceptrons**. MIT Press, Cambridge, USA, 1969.

Popovics, S. A. Numerical Approach to the Complete Stress Strain Curve Concrete. *Cement and Concrete Research*, 3(5), pp. 583-599, 1973.

Roseiro, Luís M. F. **Detecção de dano e identificação estrutural com redes neuronais artificiais**. Tese (Doutorado), Faculdade de Ciências e Tecnologia da Universidade de Coimbra, 2004;

Rosenblatt, F. 1958. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review* 6, p. 386–408, 1958.

Rumelhart, D. E., Hinton, G. E., Williams, R. J., **Learning Representations of Back-Propagation Error**. *Nature* Vol. 323, p. 533-536, 1986.

Saini, L.M.; Soni, M.K. Artificial Neural Network Based Peak Load Forecasting Using Levenberg-Marquardt and Quasi-Newton Methods. *IEE Proceedings on Generation Transmission, Distribution*, Vol.149, N.5, p. 578- 584, 2002.

Sánchez, A. C. A. **Estudo de impacto usando elementos finitos e análise não linear**. Dissertação (Mestrado), Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 126p. 2001.

Sánchez, C. A. A. "Estudo de Impacto usando Elementos Finitos e Análise Não Linear". Dissertação (mestrado) - Escola de Engenharia de São Carlos da Universidade de São Paulo, 2001.

Silva, R. C. C.; Guerreiro, J. N. C.; Loula, A. F. D. A Study of Pipe Interacting Corrosion Defects Using the FEM and Neural Networks. **Advances in Engineering Software**, Elsevier, Vol. 38, p. 868–875, 2007.

Stephens, J. E. ; Van Luchene, R. D. Integrated Assessment of Seismic Damage in Structures. **Microcomputers in Civil Engineering**, Vol. 9, p. 119-128, 1994.

Tafner, M. A. As redes neurais artificiais: aprendizado e plasticidade. **Cérebro e Mente**, São Paulo, vol. 5, 1998.

Topping, B. H. V; Bahreininejad, A. Subdomain in Generation using Parallel Q-State Potts Neural Networks. **CIVIL-COMP95** Developments in Neural Networks and Evolution-ary Computing for Civil and Structural Engineering, Topping, B. H. V. (ed.), CIVIL-COMP Press, Edinburgh, UK, 65-78, 1995.

Widrow, B.; Hoff, M. Adaptive Switching Circuits. IRE WESCON Convention Record, 1960.

Zienkiewicz O.C., Morgan K. Finite Element and Approximations. Wiley 1983.