

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ  
DEPARTAMENTO ACADÊMICO DE MECÂNICA  
CURSO DE ENGENHARIA MECÂNICA

ALISON SIEGEL  
PEDRO ARNALDO CORREA NETO

**MODELAGEM E METAMODELAGEM DE ESTRUTURA DE  
MATERIAL COMPÓSITO LAMINADO SUBMETIDO AO REGIME DE  
PÓS-FLAMBAGEM**

TRABALHO DE CONCLUSÃO DE CURSO  
(TCC 2)

CURITIBA  
2019

ALISON SIEGEL  
PEDRO ARNALDO CORREA NETO

**MODELAGEM E METAMODELAGEM DE ESTRUTURA DE  
MATERIAL COMPÓSITO LAMINADO SUBMETIDO AO REGIME DE  
PÓS-FLAMBAGEM**

Monografia do Projeto de Pesquisa apresentada à disciplina de Trabalho de Conclusão de Curso - TCC2 do curso de Engenharia Mecânica da Universidade Tecnológica Federal do Paraná, como requisito parcial para aprovação na disciplina.

Orientadora: Prof. Dra. Ana Paula Carvalho da Silva  
Ferreira

CURITIBA  
2019

## **TERMO DE APROVAÇÃO**

Por meio deste termo, aprovamos a monografia do Projeto de Pesquisa Pesquisa “MODELAGEM E METAMODELAGEM DE ESTRUTURA DE MATERIAL COMPÓSITO LAMINADO SUBMETIDO AO REGIME DE PÓS-FLAMBAGEM”, realizada pelos alunos Alison Siegel e Pedro Arnaldo Correa Neto, como requisito para aprovação na disciplina de Trabalho de Conclusão de Curso 2, do curso de Engenharia Mecânica da Universidade Tecnológica Federal do Paraná.

Prof. Dra. Ana Paula Carvalho da Silva Ferreira  
UTFPR – Damec  
Orientadora

Prof. Dr. Cláudio Tavares da Silva  
UTFPR – Damec  
Avaliador

Prof. Dr. Marco Antonio Luersen  
UTFPR – Damec  
Avaliador

Curitiba, 04 de Julho de 2019.

## RESUMO

SIEGEL, Alison, NETO, Pedro Arnaldo Correa. Modelagem e Metamodelagem de Estrutura de Material Compósito Laminado Submetido ao Regime de Pós-Flambagem. 61 f. Trabalho de Conclusão de Curso – TCC2, Bacharelado em Engenharia Mecânica, Departamento Acadêmico de Mecânica, Universidade Tecnológica Federal do Paraná. Curitiba, 2019.

A estabilidade estrutural de um projeto que envolve materiais compósitos laminados normalmente está relacionada com a determinação de sua carga crítica de flambagem. O presente trabalho visa analisar o comportamento de uma placa compósita, submetida ao regime de pós-flambagem (carregamento superior à carga crítica). A estrutura de material compósito é modelada através do software de elementos finitos Abaqus®, o qual possibilita analisar a influência de diversos parâmetros, como número de camadas e seus ângulos de orientação, na resposta estrutural. Posteriormente, através do programa MATLAB® e dos dados obtidos via elementos finitos, um metamodelo (redes neurais) é utilizado e treinado para representar a estrutura compósita, através de suas cargas e deslocamentos. A validade de metamodelo de rede neural é testada pela apresentação de novos dados de entrada, diferentes dos usados no treinamento, e posterior comparação de sua resposta com a calculada por elementos finitos. Para o presente estudo de caso, conclui-se que as redes neurais possuem uma capacidade de aproximação de resultados parcialmente satisfatória. Ou seja, ao treinar uma rede neural com as entradas de uma placa e testá-la com entradas de uma placa distinta, mas que possua comportamento mecânico (curva carga *versus* deslocamento) similar, obtém-se resultados satisfatórios. Mas ao testar a mesma rede neural com entradas de uma placa com comportamento mecânico muito distinto, ela não apresenta resultados aceitáveis.

**Palavras-chave:** Materiais compósitos, Pós-flambagem, Parâmetros de Laminação, Elementos Finitos, Análise Estrutural, Redes Neurais.

## ABSTRACT

SIEGEL, Alison, NETO, Pedro Arnaldo Correa. Modeling and Metamodeling of Laminated Composite Material Structure Submitted to the Post-Buckling Regime. 61 p. Undergraduate Thesis, Mechanical Engineering, Academic Mechanical Engineering Department, Universidade Tecnológica Federal do Paraná. Curitiba, 2019.

The structural stability of a design involving laminated composite materials is usually related to the determination of their critical buckling load. The present work aims to analyze the behavior of a composite plate, submitted to the post-buckling regime (load higher than the critical load). The composite material structure is modeled using Abaqus® finite element software, which allows the analysis of the influence of various parameters, such as number of layers and their orientation angles, on the structural response. Subsequently, through the MATLAB® program and the data obtained through finite elements, a metamodel (neural networks) is used and trained to represent the composite structure, through its loads and displacements. The validity of the neural network metamodel is tested by presenting new input data, different from those used in training, and then comparing its response with that calculated by finite elements. For the present case study, it is concluded that neural networks have a partially satisfactory approximation of results. In other words, when training a neural network with the inputs of a plate and testing it with inputs of a different one with similar mechanical behavior (load versus displacement curve), we obtain satisfactory results. However, when testing the same neural network with inputs of a plate with different mechanical behavior, it does not present acceptable results.

**Keywords:** Composite materials, Post-buckling, Lamination parameters, Finite elements, Structural analysis, Neural networks.

## LISTA DE ILUSTRAÇÕES

Figura 1 - Elemento de casca convencional comparado ao elemento de casca contínuo.....	15
Figura 2 - Gráfico de carga proporcional com resposta instável.....	18
Figura 3 - Representação das iterações do método de Riks .....	19
Figura 4 - Neurônio biológico .....	22
Figura 5 - Diagrama de bloco de um neurônio artificial com três entradas. ....	23
Figura 6 - Uma estrutura em camadas de nós para uma rede neural. ....	24
Figura 7 - Geração de dados para treinamento da rede neural .....	27
Figura 8 - Treinamento da rede neural .....	27
Figura 9 - Geometria e condições de contorno da placa analisada.....	28
Figura 10 - Malha de elementos finitos utilizada na discretização da placa. ....	29
Figura 11 - Primeiro modo de flambagem para o empilhamento [(0/90) <sub>3</sub> ] <sub>s</sub> . U3 em milímetros. .....	29
Figura 12 - Segundo modo de flambagem para o empilhamento [(0/90) <sub>3</sub> ] <sub>s</sub> . U3 em milímetros. .....	30
Figura 13 - Terceiro modo de flambagem para o empilhamento [(0/90) <sub>3</sub> ] <sub>s</sub> . U3 em milímetros. .....	30
Figura 14 - Quarto modo de flambagem para o empilhamento [(0/90) <sub>3</sub> ] <sub>s</sub> . U3 em milímetros.	31
Figura 15 - Aspecto da placa [(0/90) <sub>3</sub> ] <sub>s</sub> deformada na direção Z (U3) no regime de pós- flambagem para (a) 1% de imperfeição e (b) 10% de imperfeição. U3 em milímetros...32	32
Figura 16 - Validação do modelo para o laminado [(0/90) <sub>3</sub> ] <sub>s</sub> com 1% de imperfeição. ....	33
Figura 17 - Validação do modelo para o laminado [(0/90) <sub>3</sub> ] <sub>s</sub> com 10% de imperfeição. ....	34
Figura 18 - Definição da amostra da rede.....	36
Figura 19 - Número de neurônios na camada oculta da rede .....	36
Figura 20- Resultados das análises no Abaqus® para placas com 10% de imperfeição. Placas 10 e 17 apresentadas em vermelho e preto respectivamente .....	38

Figura 21 - Análise de regressão da rede neural treinada para a placa 10 com 10% de imperfeição .....	39
Figura 22 - Comparação da saída da rede neural treinada para a placa 10 com os resultados obtidos via Abaqus® .....	39
Figura 23 - Análise de regressão da rede neural testada para a placa 17 com 10% de imperfeição .....	40
Figura 24 - Comparação dos resultados da rede neural testada para a placa 17 com resultados obtidos via Abaqus® .....	40
Figura 25 - Resultados das análises no Abaqus® para placas com 1% de imperfeição. Placas 10 e 17 apresentadas em vermelho e preto respectivamente .....	41
Figura 26- Análise de regressão da rede neural treinada para a placa 10 com 1% de imperfeição .....	42
Figura 27 - Comparação da saída da rede neural treinada para a placa 10 com os resultados obtidos via Abaqus® .....	42
Figura 28 - Análise de regressão da rede neural testada para a placa 17 com 1% de imperfeição .....	43
Figura 29 - Comparação dos resultados da rede neural testada para a placa 17 com resultados obtidos via Abaqus® .....	43
Figura 30 - Resultados das análises no Abaqus® para placas com 10% de imperfeição. Placas 10 e 15 apresentadas em vermelho e preto respectivamente .....	44
Figura 31 - Análise de regressão da rede neural testada para a placa 15 com 10% de imperfeição .....	45
Figura 32 - Comparação dos resultados da rede neural testada para a placa 15 com resultados obtidos via Abaqus® .....	45
Figura 33 - Resultados das análises no Abaqus® para placas com 1% de imperfeição. Placas 10 e 15 apresentadas em vermelho e preto respectivamente .....	46
Figura 34 - Análise de regressão da rede neural testada para a placa 15 com 1% de imperfeição .....	47
Figura 35 - Comparação dos resultados da rede neural testada para a placa 15 com resultados obtidos via Abaqus® .....	47

Figura 36 - Comparação entre placas modeladas com 1% de imperfeição.....	48
Figura 37 - Análise de regressão da rede neural testada para a placa 5 com 1% de imperfeição .....	49
Figura 38 - Comparação dos resultados da rede neural testada para a placa 5 com resultados obtidos via Abaqus® .....	49
Figura 39 - Análise de regressão da rede neural testada para a placa 8 com 1% de imperfeição .....	50
Figura 40 - Comparação dos resultados da rede neural testada para a placa 8 com resultados obtidos via Abaqus® .....	50
Figura 41 - Análise de regressão da rede neural testada para a placa 4 com 1% de imperfeição .....	51
Figura 42 - Comparação dos resultados da rede neural testada para a placa 4 com resultados obtidos via Abaqus® .....	51
Figura 43 - Análise de regressão da rede neural testada para a placa 19 com 1% de imperfeição .....	52
Figura 44 - Comparação dos resultados da rede neural testada para a placa 19 com resultados obtidos via Abaqus® .....	52
Figura 45 - Análise de regressão da rede neural testada para a placa 19 com 1% de imperfeição com alteração dos parâmetros de treinamento.....	54
Figura 46 - Comparação dos resultados da rede neural testada para a placa 19, com novos parâmetros de treinamento, com resultados obtidos via Abaqus® .....	54
Figura 47 - Geometria do modelo utilizado.....	57
Figura 48 - Sistema de coordenadas de referência para definir as direções locais do material	58
Figura 49 - Seções criadas para definir os conjuntos de geometrias .....	58
Figura 50 - Forma deformada e resultado obtido na análise linear .....	60
Figura 51 - Forma deformada e não deformada da placa analisada não-linearmente .....	60
Figura 52 - Deslocamento nas extremidades da região central da placa não-linear.....	61



## LISTA DE TABELAS

Tabela 1 - Dados da lâmina de material AS4/9310 .....	28
Tabela 2 - Valores de carga de bifurcação para o empilhamento $[(0/90)_3]_s$ em N/mm. ....	31
Tabela 3 - Fatores de escala utilizados para os respectivos modos de flambagem. ....	32
Tabela 4 - Configurações angulares obtidas através do DoE .....	35
Tabela 5 - Comparação da análise de regressão (R).....	48
Tabela 6 - Regressão (R) do teste das redes treinadas com diferentes configurações de parâmetros da <i>toolbox</i> .....	53

# SUMÁRIO

1	INTRODUÇÃO	11
1.1	Contexto do Tema	11
1.2	Caracterização do problema	12
1.3	Objetivos	12
1.3.1	Objetivos gerais	12
1.3.2	Objetivos específicos	13
1.4	Justificativa	13
2	FUNDAMENTAÇÃO TEÓRICA	14
2.1	Pós-flambagem	14
2.2	Pacote de programas Abaqus® FEA®	14
2.2.1	Elementos de casca	15
2.2.2	Análise de flambagem por autovalores	16
2.2.3	Método de Riks para pós-flambagem	17
2.2.4	Imperfeições	20
2.3	Redes Neurais	21
3	METODOLOGIA	25
3.1	Descrição da metodologia	25
4	RESULTADOS	28
4.1	Análise de placas compósitas no Abaqus®	28
4.1.1	Análise de placas compósitas perfeitas sob carregamento compressivo	28
4.1.2	Análise de placas compósitas imperfeitas sob carregamento compressivo	32
4.2	Definição e modelagem de 20 placas com diferentes orientações angulares	34
4.3	Resultados Redes Neurais	35
4.3.1	Comparação entre a placa 10 e a placa 17 com imperfeição de 10%	37
4.3.2	Comparação entre a placa 10 e a placa 17 com imperfeição de 1%	41
4.3.3	Comparação entre a placa 10 e a placa 15 com imperfeição de 10%	44
4.3.4	Comparação entre a placa 10 e a placa 15 com imperfeição de 1%	46
4.3.5	Comparação da placa 10 com as placas 5, 8, 4 e 19, com 1% de imperfeição	48
4.3.6	Modificação dos parâmetros da <i>toolbox</i> de redes neurais.	53
5	Considerações finais	55
	REFERÊNCIAS	56
	APÊNDICE A - ANÁLISE NÃO LINEAR DE UMA PLACA ISOTRÓPICA INCLINADA	57

# 1 INTRODUÇÃO

## 1.1 Contexto do Tema

Uma estrutura compósita é um sistema composto de duas ou mais fases, cuja performance e as propriedades mecânicas são projetadas para serem superior à dos materiais que o constituem agindo independentemente. Uma das fases é normalmente descontínua, rígida e mais resistente, chamada de fibra. A outra fase, de menor rigidez e resistência é chamada de matriz (DANIEL, et al., 1994)

Um material compósito laminado é feito de duas ou mais lâminas de material empilhados em diversas orientações angulares. Devido a sua alta rigidez e resistência mecânica, existem diversas pesquisas relacionadas a aplicação de materiais compósitos laminados. No entanto, esse tipo de material é altamente sensível a imperfeições geométricas, o que torna o seu comportamento de flambagem e pós-flambagem um importante critério de projeto.

Através da determinação de uma combinação de parâmetros, como orientação angular e número de camadas, é possível projetar estruturas, que submetidas a cargas mais elevadas, apresentam redução de peso e deflexões menores.

No presente estudo, uma placa de material laminado é modelada via elementos finitos. Em seguida, são testadas diversas combinações de parâmetros de orientações angulares para as camadas do laminado. As respostas estruturais, cargas e deslocamentos, são armazenadas para posterior treinamento do metamodelo.

A determinação da melhor orientação angular dos laminados demanda bastante tempo e esforço computacional, mesmo quando um método de otimização é utilizado. Metamodelos podem ser usados em processos de otimização como substitutos do modelo original, com o propósito de reduzir o tempo e o custo computacional dessa otimização. Dentre as opções para metamodelos estão as redes neurais artificiais.

As redes neurais são modelos matemáticos que através de um devido treinamento são capazes de simular um fenômeno e apresentar resultados próximos dos encontrados através de uma formulação analítica ou numérica. Ou seja, ao fornecermos os parâmetros de orientação angular como entrada e as cargas, os quais foram simulados via elementos finitos, esperamos treinar o metamodelo para que ele seja capaz de retornar saídas, deformações, próximas as que foram encontrados via elementos finitos. Dessa forma testaremos a eficiência das redes neurais em estimar o comportamento de uma estrutura compósita no regime de pós-flambagem.

## 1.2 Caracterização do problema

Ao se projetar um material compósito laminado, busca-se encontrar uma combinação de parâmetros que possibilitem o desenvolvimento de um material com menor peso e menor deflexão.

No entanto, normalmente, essa otimização demanda custos e tempo computacional elevados, mesmo quando feita através de um algoritmo de otimização.

Tendo isso em mente, esse estudo consiste em modelar por elementos finitos uma placa compósita no regime de pós-flambagem. O regime de pós-flambagem é estabelecido através da aplicação de um carregamento pré-determinado acima da carga crítica. Para compor uma base de dados para o treinamento da rede neural, as orientações angulares das camadas dos laminados são alteradas e registra-se os correspondentes deslocamentos, referentes à carga aplicada. Deste modo, a rede neural recebe orientações angulares e cargas como entrada e deslocamentos como saídas desejadas. Após seu treinamento será possível avaliar a eficiência de uma rede neural na metamodelagem de uma estrutura compósita no regime de pós-flambagem. A validação dessa rede neural, como representativa da estrutura, possibilita sua inserção em processos de otimização, reduzindo o custo computacional.

## 1.3 Objetivos

### 1.3.1 Objetivos gerais

O objetivo principal deste trabalho é realizar a modelagem e metamodelagem de uma placa de material compósito laminado no regime de pós-flambagem. Para isto, a placa é modelada por elementos finitos, utilizando o programa computacional Abaqus®. O modelo de elementos finitos é validado por comparação com a literatura.

Após a validação do modelo, realiza-se o treinamento de um metamodelo de redes neurais através do programa MATLAB®. O qual tem como objetivo aproximar os resultados, deslocamentos, encontrados via elementos finitos. Após o treinamento, novas entradas são apresentadas à rede neural e verifica-se a compatibilidade entre sua resposta e a resposta por elementos finitos. Havendo boa compatibilidade pode-se concluir que a rede treinada representa a estrutura em questão.

### 1.3.2 Objetivos específicos

Os objetivos específicos são:

- Estudar o método de elementos finitos;
- Modelar placa compósita escolhida para análise, utilizando o programa Abaqus®;
- Fazer a validação dos resultados da modelagem da placa, utilizando dados encontrados na literatura;
- Estudar redes neurais;
- Realizar o treinamento do metamodelo, visando resultados que se aproximem dos obtidos via elementos finitos, através do programa computacional MATLAB®;
- Testar a capacidade de aproximação do metamodelo de rede neural pela apresentação de novas entradas e comparação de sua saída com a saída desejada.

### 1.4 Justificativa

O presente trabalho de análise estrutural tem por objetivo integrar conhecimentos obtidos até agora na graduação de Engenharia Mecânica. No decorrer de seu desenvolvimento, é necessário retomar e aprimorar conhecimentos nas áreas de materiais, mecânica estrutural, métodos numéricos, entre outras.

Ademais, em se tratando da inevitabilidade da resolução de problemas cada vez mais complexos, torna-se necessário melhor familiaridade com ferramentas computacionais de engenharia. Este trabalho propõe a solução para um problema relacionado a um material compósito laminado, através de uma abordagem computacional não-linear. Abordagem esta que envolve análise de elementos finitos no software Abaqus® e o treinamento de uma rede neural no software MATLAB®.

Por fim, os envolvidos neste trabalho têm o intuito de proporcionar aos alunos de engenharia mecânica da UTFPR um maior conhecimento na área de mecânica estrutural, bem como disponibilizar uma metodologia de utilização dos dois programas computacionais citados acima. E assim sendo, prestar auxílio para que eles possam se tornar profissionais mais capacitados, tanto na indústria quanto na academia.

## 2 FUNDAMENTAÇÃO TEÓRICA

### 2.1 Pós-flambagem

Estruturas de materiais compósitos laminados possuem diversas aplicações devido a suas elevadas rigidezes e resistências mecânicas em relação ao peso. No entanto, normalmente essas estruturas são finas e sujeitas a carregamentos compressivos, o que as tornam bastante suscetíveis à flambagem.

Com o intuito de obter uma estrutura mais leve e confiável, análises do comportamento de flambagem da estrutura são frequentemente realizadas. Contudo, uma redução ainda mais significativa do peso é possível se a estrutura for apta a trabalhar no regime de pós-flambagem (carregamento acima da carga crítica de flambagem).

Projetos que seguem as recomendações de resistência à flambagem, através de modelos analíticos lineares e fatores de correção empíricos, normalmente atingem um padrão aceitável de confiabilidade. Mas em algumas situações eles fornecem resultados demasiadamente conservadores. Através do avanço computacional, foi possível realizar análises de modelos numéricos não-lineares. O que permitiu estudos de otimização estrutural de placas compósitas em regime de pós-flambagem, por conseguinte tornou-se possível projetar estruturas laminadas com melhores desempenhos (FALZON, et al., 2008).

### 2.2 Pacote de programas Abaqus® FEA®

Abaqus® FEA® é um pacote de programas para análise de elementos finitos (FEA - *finite element analysis*) e engenharia auxiliada por computador (CAE - *computer aided engineering*) criado em 1978 (originalmente como ABAQUS®), sendo adquirido pela DassaultSystèmes® em 2005, empresa detentora de outros programas conhecidos no mercado, como CATIA® e SOLIDWORKS®. Atualmente, este é constituído de cinco módulos, são eles: Abaqus® CAE (*Complete Abaqus® Environment*), Abaqus® Standard, Abaqus® Explicit, Abaqus® CFD e Abaqus® Electromagnetic.

No presente trabalho, será empregado somente o módulo Abaqus® CAE®. Tal programa tem como recursos um ambiente de modelagem e montagem de componentes mecânicos em geral, definição de propriedades dos materiais empregados, geração de malha de elementos finitos, e permite a visualização da análise realizada e dos dados de saída.

As informações abordadas a seguir são encontradas no manual do software Abaqus®, elas são importantes para definir e determinar alguns parâmetros durante o processo de modelagem.

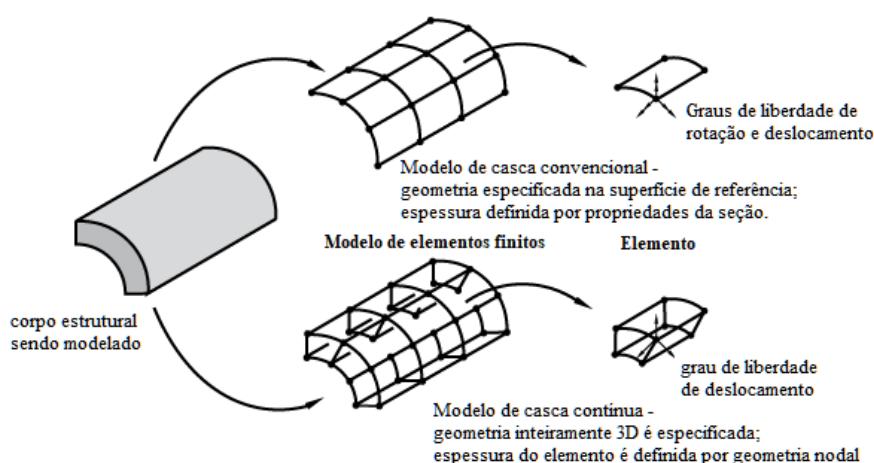
### 2.2.1 Elementos de casca

Elementos de casca são modelos de estruturas, nos quais uma dimensão (espessura) é significativamente menor que outras dimensões. Estes são divididos em elementos de casca convencionais e elementos de casca contínuos.

Elementos de casca convencionais utilizam essa condição para discretizar um corpo através da definição da geometria em uma superfície de referência. Nesse caso a espessura é definida através da seção “definições de propriedades”. São elementos bidimensionais e possuem deslocamentos e rotações como grau de liberdade.

Diferentemente, elementos de casca contínuos discretizam um corpo inteiramente tridimensional. A espessura é determinada pela geometria nodal do elemento. Por um ponto de vista de modelagem, elementos de casca contínuos se assemelham com um sólido contínuo tridimensional, mas a sua cinemática e o seu comportamento são similares a de um elemento de casca convencional. Estes elementos possuem apenas deslocamentos como grau de liberdade.

**Figura 1 - Elemento de casca convencional comparado ao elemento de casca contínuo**



Fonte: Adaptado de (Dassault Systèmes®, 2014)

As direções locais padrão do Abaqus®, usadas nas superfícies da casca para definição de propriedades de materiais e para informar tensão e deformação dos componentes, são definidas por convenções do software. É possível alterar essas convenções e definir outras direções locais de orientação.

### 2.2.2 Análise de flambagem por autovalores

A análise por autovalores é um procedimento de perturbação linear, utilizado geralmente para estimar a carga crítica de flambagem ( $P_{\text{crítico}}$ ) de estruturas de alta rigidez. Estas estruturas, submetidas ao regime de flambagem, tem suas cargas de projeto orientadas axialmente ou distribuídas, ao invés de ação de flexão. Suas respostas geralmente envolvem pouca deformação antes da ocorrência da flambagem.

A flambagem é um fenomeno verificado em estruturas esbeltas (área da seção transversal pequena em comparação ao seu comprimento), que ao sofrer um esforço compressivo axial que ultrapasse a carga crítica de flambagem colapsam, dobrando-se subitamente.

Um exemplo de tal fenômeno é a coluna Euler. A qual representa uma estrutura extremamente rígida submetida à uma carga axial compressiva, que aumenta gradativamente até que uma carga crítica seja atingida. Neste momento, a estrutura dobra-se repentinamente, exibindo uma rigidez menor. Porém, mesmo quando a resposta de uma estrutura é não-linear antes do colapso, uma análise do processo de flambagem por autovalores pode fornecer estimativas das cargas críticas e dos modos de colapso. Desta forma, são úteis na análise em pós-flambagem. Tal análise pode ser utilizada também na investigação da sensibilidade a imperfeições.

O programa utilizado, Abaqus®, oferece os métodos de Lanczos e de iteração por subespaço para a resolução de tal problema. O método de Lanczos é, geralmente, mais rápido quando muitos modos de autovalor são requeridos para um sistema com vários graus de liberdade. Já o método de iteração por subespaço se mostra mais eficiente e rápido na resolução, quando menos de vinte modos de autovalor são requeridos. Assim sendo, seleciona-se este último método para o desenvolvimento das análises deste trabalho.

Para ambos os métodos de solução, são especificados o número desejado de autovalores, e o próprio software escolherá um número de vetores para a iteração por subespaço.

Um padrão de incremento ( $Q^N$ ) é definido na etapa de predeterminação do autovalor de flambagem. A magnitude desse carregamento não possui elevada importância, uma vez que será colocada em escala pelos multiplicadores de carga ( $\lambda_i$ ), encontrados no problema de autovalor.



$$(\mathbf{K}_0^{NM} + \lambda_i \mathbf{K}_\Delta^{NM}) \mathbf{v}_i^M = \mathbf{0} \quad (1)$$

Onde:

- $K_0^{NM}$  é a matriz de rigidez correspondente ao estado base, que inclui os efeitos das pré-cargas  $P^N$ , se existentes;
- $K_\Delta^{NM}$  é o diferencial de tensão inicial e a matriz de rigidez de carga devido ao padrão de carga incremental  $Q^N$ ;
- $\lambda_i$  são os autovalores, que representam possíveis fatores multiplicativos para determinar o carregamento crítico;
- $v_i^M$  são os modos de flambagem ou autovetores;
- $M$  e  $N$  referem-se aos graus de liberdade do modelo completo e;
- $i$  refere-se ao  $i$ -ésimo modo de flambagem.

As cargas críticas de flambagem são expressas por  $P^N + \lambda_i Q^N$ . Geralmente, o menor valor de  $\lambda_i$  obtido nas análises é o valor de interesse. O padrão de pré-carga ( $P^N$ ) e o padrão de carga de perturbação ( $Q^N$ ), podem ser diferentes. Para ilustrar,  $P^N$  pode ser diferente devido a fatores térmicos causados por mudanças de temperatura, enquanto  $Q^N$  é causado apenas pela aplicação de pressão.

Os modos de flambagem ( $v_i^M$ ) são vetores normalizados e não representam a magnitude real das deformações à uma determinada carga crítica. Eles são normalizados tal que o máximo deslocamento do componente é 1,0. Se todas as componentes do deslocamento forem iguais a zero, a máxima rotação do componente é normalizada para 1,0. As formas de modos de flambagem são geralmente a maior vantagem da aplicação da análise por autovalores, uma vez que ela prevê o provável modo de falha da estrutura.

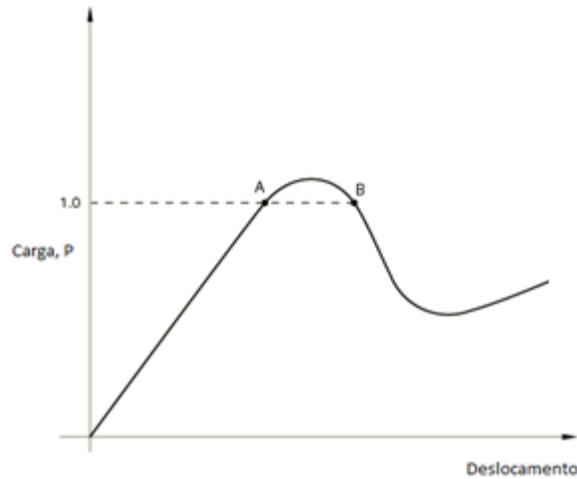
### 2.2.3 Método de Riks para pós-flambagem

Em casos simples, a análise linear por autovalores é suficiente para a avaliação de projeto. Porém, se há preocupação quanto a não-linearidade do material, não-linearidade geométrica antes da flambagem ou pós-flambagem instável, o método de Riks deve ser utilizado para avaliar o problema.

Este método é apresentado em RIKS (1979) e, é empregado para casos em que o carregamento é proporcional, ou seja, onde as grandezas de carga são governadas por um único

parâmetro escalar. O método pode fornecer soluções mesmo em casos de respostas instáveis e complexas, como ilustrado na Figura 2.

**Figura 2 - Gráfico de carga proporcional com resposta instável**



Fonte: Adaptado de (Dassault Systèmes®, 2014)

O método de Comprimento de Arco ou método de Riks é um método eficiente para resolução de problemas não-lineares quando este exibe um ou mais pontos críticos. No caso de um carregamento de pós-flambagem, um ponto crítico pode ser descrito como o ponto em que a estrutura carregada não suporta o aumento das forças externas atuantes, tornando-se instável.

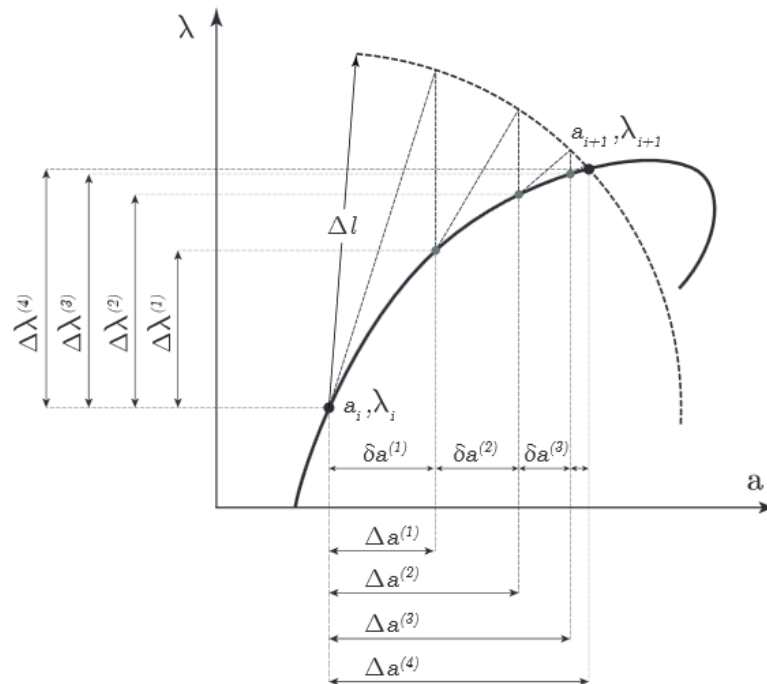
O sistema de equações não-lineares para este problema é dado na forma:

$$\mathbf{F}^{int}(\mathbf{u}) - \mathbf{F}^{ext} = \mathbf{0} \Rightarrow \mathbf{F}^{int}(\mathbf{u}) - \lambda \mathbf{q} = \mathbf{0} \quad (2)$$

Onde:

- $\mathbf{F}^{int}(\mathbf{u})$  é o vetor de carga interna em função do campo de deslocamento  $\mathbf{u}$ ;
- $\mathbf{u}$  é o campo de deslocamento;
- $\mathbf{F}^{ext}$  é o vetor de carga externa que, como sua direção permanece constante durante a análise e somente sua magnitude é alterada, pode ser reescrito como uma constante  $\mathbf{q}$  e ter a variação controlada através do fator  $\lambda$ ;
- $\lambda$  é o incremento da carga ou LPF (*Load Proportionality Factor*);
- $\mathbf{q}$  é um valor de carga constante durante a análise, a fim de simplificar o sistema de equações.

Figura 3 - Representação das iterações do método de Riks



Fonte: (VASIOS, 2015)

Na Figura 3, o parâmetro  $a$  representa um deslocamento normalizado, enquanto  $\lambda$  o incremento da carga ou LPF (*Load Proportionality Factor*). Tal incremento é definido pelo raio do círculo  $\Delta l$  e o próximo ponto é o ponto de intersecção entre o caminho e o círculo de raio  $\Delta l$ . Contudo, a maneira de formular o método de Riks descrita acima não é a mais eficiente. Como resultado, os *softwares* comerciais utilizam uma abordagem diferente para esse método, chamando-o de método modificado de Riks.

Outra razão é que muitos materiais (e possíveis cargas) de interesse tem respostas dependente do caminho, sendo assim necessário limitar o tamanho do incremento. No método modificado de Riks, como implementado no Abaqus®, o tamanho do incremento é limitado movendo uma determinada distância ao longo da linha tangente ao ponto de solução atual. Em seguida, procura-se o equilíbrio no plano que passa pelo ponto obtido e que é ortogonal à mesma linha tangente.

No Abaqus®, caso a etapa de Riks seja uma continuação de um histórico anterior, quaisquer cargas que existam no início da etapa e que não são redefinidas, tem suas respectivas magnitudes constantes e tratadas como “pré-cargas”. Uma carga definida na etapa de Riks é tida como pré-carga e todas as cargas prescritas iniciam do valor da “carga morta” ( $P_0$ ) até os valores de referência especificados ( $P_{ref}$ ).

Dentro da etapa de Riks, a carga atual ( $P_{total}$ ) é sempre proporcional, respeitando a seguinte equação:

$$\mathbf{P}_{\text{total}} = \mathbf{P}_0 + \lambda (\mathbf{P}_{\text{ref}} - \mathbf{P}_0), \quad (3)$$

$\lambda$  o fator de proporcionalidade de carga ou LPF (*Load Proportionality Factor*). Como mencionado anteriormente e, é parte da solução e é dado ao final de cada incremento.  $\lambda$  em conjunto com o comprimento do arco ( $l$ ) ao longo do caminho de equilíbrio estático no espaço de deslocamento da carga, fornecem as soluções independentemente de a resposta ser estável ou instável.

#### 2.2.4 Imperfeições

Imperfeições são geralmente introduzidas por perturbações na geometria. O software Abaqus® oferece três maneiras para definição de uma imperfeição: pela superposição linear dos modos de flambagem obtidos em uma análise prévia por autovalores; a partir dos deslocamentos de uma análise estática; ou ainda especificando diretamente o número do nó e os valores das respectivas imperfeições. A menos que a forma precisa de uma imperfeição seja conhecida, uma imperfeição consistindo em múltiplos modos de flambagem sobrepostos pode ser introduzida.

A abordagem usual engloba duas análises com a mesma definição de modelo. Sendo a primeira a fim de estabelecer os modos de colapso prováveis e, a segunda, para realizar a análise de pós-flambagem.

A primeira análise consiste em uma análise de autovalores com o Abaqus® na estrutura considerada “perfeita” para estabelecer os prováveis modos de colapso e verificar se a malha descreve esses modos com precisão.

Na segunda análise, as imperfeições são introduzidas na geometria adicionando esses modos de flambagem à geometria “perfeita”. Os modos de flambagem mais baixos são geralmente considerados os que fornecem as imperfeições mais críticas, então normalmente eles são dimensionados e adicionados à geometria perfeita para criar a malha perturbada. A imperfeição tem assim a forma

$$\Delta \mathbf{x}_i = \sum_{i=1}^M \omega_i \boldsymbol{\varphi}_i, \quad (4)$$

onde  $\boldsymbol{\varphi}_i$  é o  $i$ -ésimo modo e  $\omega_i$  é o fator de escala associado.

Os fatores de escala para os vários modos utilizados devem ser escolhidos. Geralmente, se a estrutura não for sensível a imperfeições, o menor modo de flambagem deve ter o maior fator de escala. As grandezas das perturbações utilizadas são, em geral, um

percentual de uma dimensão da estrutura. Por exemplo, a seção transversal de uma viga ou a espessura em um elemento de casca.

Após a definição das imperfeições, a análise de pós-flambagem pode ser conduzida através da execução de uma análise não-linear da estrutura contendo as imperfeições, empregando o método de Riks. Assim, o método de Riks pode ser usado para executar análises de pós-flambagem de estruturas rígidas que mostram comportamento linear antes flambar. Ao realizar uma análise de carga-deslocamento, outros efeitos não-lineares importantes, como a inelasticidade do material ou contato, podem ser incluídos.

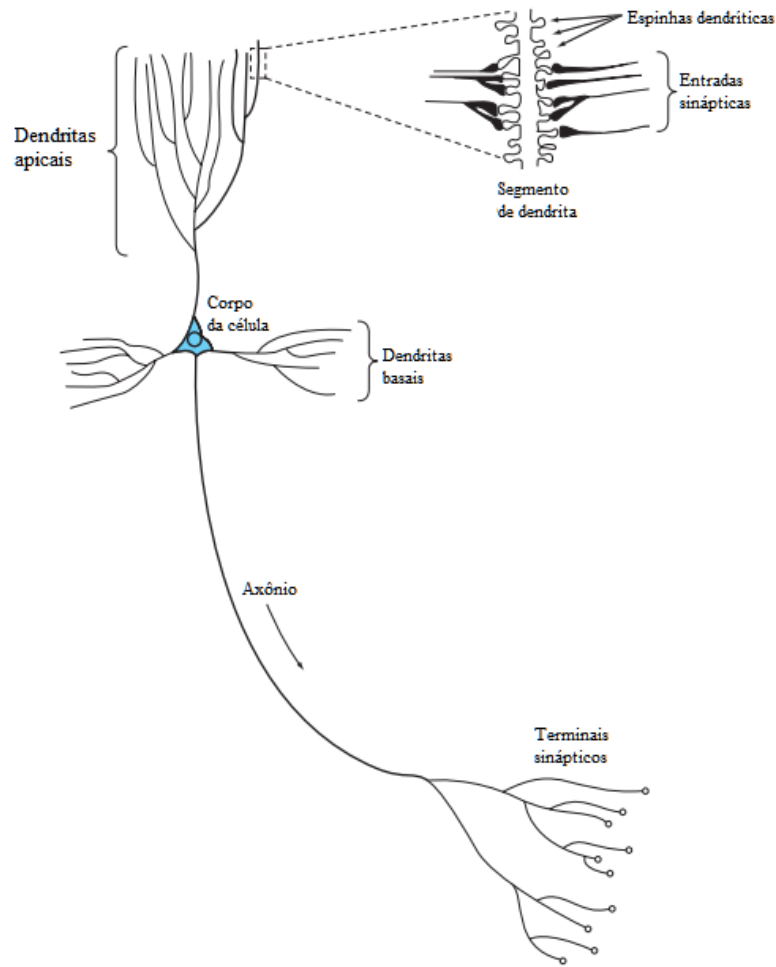
Perturbações em cargas ou condições de contorno também podem ser usadas para introduzir imperfeições iniciais. Nesse caso, pode-se usar cargas fictícias para iniciar a instabilidade. Normalmente essas cargas são aplicadas como pré-cargas antes da análise pelo método de Riks, de modo que tenham magnitudes fixas. As magnitudes dessas cargas devem ser suficientemente pequenas para que não afetem a solução de pós-flambagem.

### 2.3 Redes Neurais

De maneira habitual, a modelagem de uma resposta estrutural deve levar em consideração equações que determinam o comportamento do fenômeno, bem como condições de contorno e coeficientes que definem as propriedades do material analisado. Desta forma, as redes neurais artificiais funcionam como uma alternativa bastante interessante. Pois realizando manipulações matemáticas, que dependem apenas da entrada-saída do sistema, e desde que bem treinadas podem representar o modelo estrutural de maneira satisfatória. O modelo matemático das redes neurais faz analogia com a estrutura de funcionamento de um neurônio biológico. O seu treinamento é supervisionado por algoritmo, em que o usuário seleciona conforme a aplicação desejada. No presente trabalho utilizou-se o algoritmo de Levenberg-Marquardt.

Os neurônios biológicos são divididos em três sessões: corpo da célula (ou soma), dendritos e axônio. O corpo mede alguns milésimos de milímetro e os dendritos apresentam poucos milímetros. O axônio pode ser mais longo que as demais componentes do neurônio. Os dendritos têm a função de receber as informações ou *impulsos elétricos* processados e fornecidos por outros neurônios através do axônio. A região de contato entre a terminação axônica de um neurônio e os dendritos de outro neurônio é chamada de *sinapse*. É pelas sinapses que os neurônios se unem funcionalmente, formando redes neurais. As sinapses funcionam como válvulas, e são capazes de controlar a transmissão de impulsos - isto é, o fluxo da informação - da rede neural, (KOVÁCS, 2002).

**Figura 4 - Neurônio biológico**

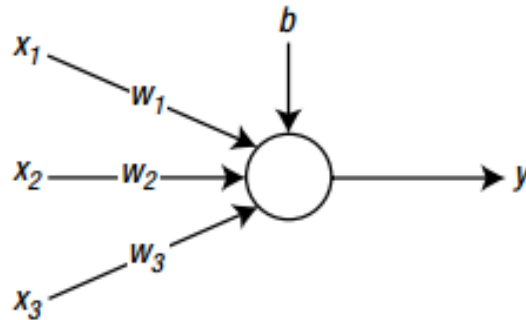


**Fonte: Adaptado de (HAYKIN, 2009)**

Os neurônios artificiais se inspiram no mecanismo de funcionamento dos neurônios biológicos, Figura 5. Dentro de um modelo de rede neural, uma série de sinapses ou conexões entre neurônios são caracterizadas por pesos. Desta maneira, um sinal  $x_j$  na entrada de uma sinapse  $j$  conectada a um neurônio  $k$  é multiplicado por um peso sináptico  $w_{k,j}$ . O primeiro subscrito refere-se ao neurônio em questão e o segundo subscrito refere-se à entrada final da sinapse à que o peso se refere. Diferentemente de um neurônio biológico, os pesos de uma rede neural artificial podem contemplar uma faixa que inclui valores negativos e positivos.

Além disso, um modelo neural também inclui uma aplicação externa, denominada *bias*  $b_i$ . As *bias* têm o efeito de aumentar ou diminuir a entrada da *função de ativação* da rede neural dependendo se o seu sinal é positivo ou negativo.

Figura 5 - Diagrama de bloco de um neurônio artificial com três entradas.



Fonte: (KIM, 2017)

Em termos matemáticos, pode-se descrever um neurônio  $k$  através das seguintes equações:

$$u_k = \sum_{j=1}^R W_{kj} x_j \quad (5)$$

e

$$y_k = \varphi(u_k + b_k) \quad (6)$$

Onde,

- $x_1, x_2, \dots, x_m$  são sinais de entrada;
- $w_{k1}, w_{k2}, \dots, w_{km}$  são os pesos sinápticos do neurônio  $k$ ;
- $u_k$  é a saída linear combinada devido ao sinal de entrada;
- $b_k$  é a *bias*;
- $\varphi$  é a função de ativação;
- $y_k$  é o sinal de saída do neurônio.

O uso de *bias*  $b_k$  tem o efeito de aplicar uma transformação a fim à saída  $u_k$  do modelo de combinação linear da Figura 5.

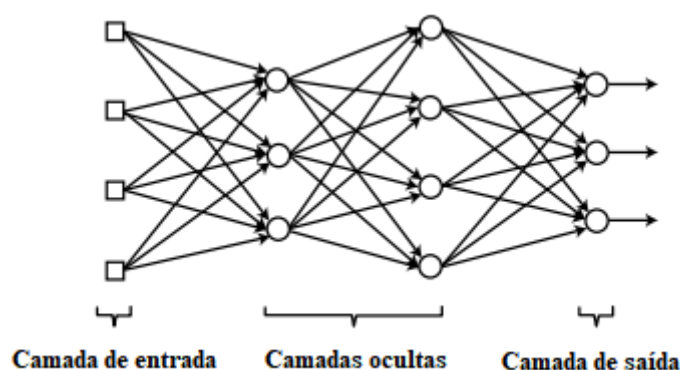
A *bias*  $b_k$  é um parâmetro externo de um neurônio artificial  $k$ . Sua entrada é

$$x_0 = 1 \quad (7)$$

e o peso é:

$$w_{k0} = b_k \quad (8)$$

**Figura 6 - Uma estrutura em camadas de nós para uma rede neural.**



Fonte: (KIM, 2017)

Uma rede neural é composta por camadas de neurônios. Existem redes neurais de camada única e de multicamada. As redes neurais de camada única possuem uma estrutura matemática, a qual define que os neurônios que fornecem as saídas são os mesmos que recebem as entradas. Em problemas de classificação de padrões, redes neurais de camada única são amplamente utilizadas. Nestes casos, o número de neurônios presentes na camada era diretamente proporcional ao número de categorias em que um padrão poderia ser classificado.

Já as redes neurais multicamada, Figura 6, são aquelas que apresentam uma ou mais camadas de neurônios entre a entrada e a saída. Estas são designadas como camadas ocultas ou intermediárias, e a saída de uma camada oculta é a entrada da camada subsequente (FERREIRA, 2005).

As redes neurais multicamadas, são úteis em diversas aplicações. Existem casos em que a alteração do número de camadas ocultas melhora significativamente o desempenho da rede e sua capacidade de generalização.



### 3 METODOLOGIA

#### 3.1 Descrição da metodologia

A análise estrutural via Método de Elementos Finitos é realizada através de discretização de um domínio em subdomínios. Permite encontrar soluções bastante próximas da exata para geometrias complexas, que não possuem solução analítica. Além disso, a utilização de um pacote comercial, o programa Abaqus®/CAE®, elimina a necessidade de programação do método numérico.

A escolha pela análise de uma estrutura compósita no regime de pós-flambagem está alinhada com o crescente uso desses materiais na indústria aeronáutica e automobilística. Materiais compósitos possuem boas propriedades mecânicas com baixo peso, ainda mais quando no regime de pós-flambagem.

Uma desvantagem de análises por elementos finitos é que elas podem apresentar alto custo computacional. Pensando ainda que o cálculo de uma resposta estrutural por elementos finitos pode ser a função objetivo de um processo de otimização, esse alto custo computacional pode inviabilizar o projeto. Os metamodelos surgem como uma opção em termos redução de custo computacional. Quando bem treinados, conseguem aproximar a resposta do modelo de forma satisfatória com menor custo.

As redes neurais artificiais são o metamodelo escolhido para o trabalho, pois de forma geral, apresentam bom desempenho em suas aplicações. Além disso, o programa MATLAB® possui uma *toolbox* de redes neurais, com interface bastante amigável ao usuário. A análise de seu desempenho na representação da estrutura proposta poderá ser avaliada de forma rápida com o auxílio desse programa.

A primeira parte do trabalho tem como objetivo desenvolver um modelo estrutural via elementos finitos, utilizando o software Abaqus®, e submetê-lo ao regime de pós-flambagem. Em seguida, comparar os resultados, curva carga vs deformação, com a literatura a fim de validar as respostas obtidas.

A análise se baseia fundamentalmente em submeter uma placa fina, de material compósito com carregamento compressivo, ao regime de flambagem, onde se observa um comportamento linear, e posteriormente através da inclusão de efeitos de não-linearidade ao estudo, submetê-lo ao regime de pós-flambagem.

O próprio manual do *software* (Dassault Systèmes®, 2014), no capítulo 8 da seção *Getting Started with Abaqus®: Interactive Edition*, apresenta uma descrição das diferenças

entre análise estrutural linear a não-linear e a forma de implementá-la no Abaqus®. Portanto, a etapa inicial desse projeto tem como ênfase a inserção de efeitos de não-linearidade em uma análise estrutural via elementos finitos, utilizando o programa Abaqus®, submetendo-a a um regime de pós-flambagem. Esta análise está disponível no Apêndice A do presente trabalho, em forma de tutorial.

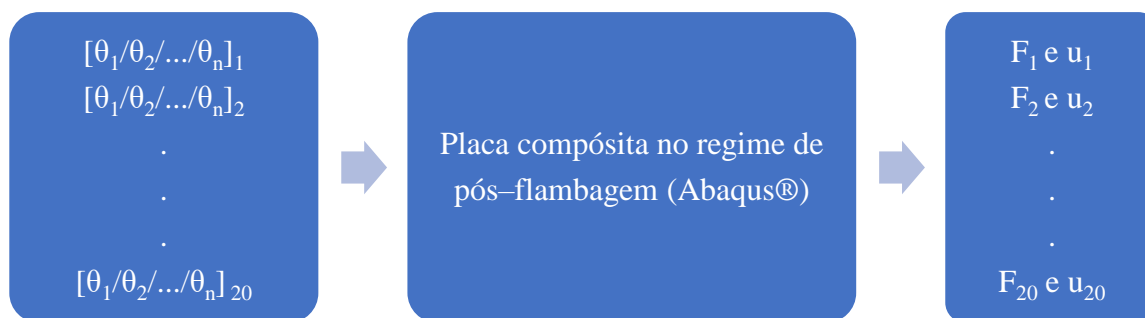
Após a inclusão dos efeitos de não-linearidade ao modelo, e por consequência, submetê-lo ao regime de pós-flambagem, faz-se a sua validação, através de comparação com modelo apresentado na literatura.

A fim de ampliar o campo amostral do estudo, um total de 20 novas placas compósitas são modeladas Abaqus®. O laminado é composto por doze camadas, divididas simetricamente. As placas são configuradas por diferentes orientações angulares ( $\theta_i$ ,  $i = 1$  a 12) de suas camadas. Sendo assim, definiu-se seis opções angulares distintas para cada uma das camadas, podendo se repetir no laminado. As orientações empregadas são:  $0^\circ$ ,  $\pm 30^\circ$ ,  $\pm 45^\circ$  ou  $90^\circ$ . A fim de definir os 20 empilhamentos que melhor representam o campo amostral, é utilizado o a ferramenta de DoE (*Design of Experiments*) para realizar um planejamento fatorial com número de fatores igual a seis.

Cada placa compósita é simulada no regime de pós-flambagem. Para isto, aplica-se um carregamento compressivo crescente  $F$  e a inserção de imperfeições ao material. Desta forma, calcula-se o deslocamento transversal  $u$  de forma a se obter um gráfico  $F$  vs  $u$  como resposta.

Para a rede neural, compõe-se uma matriz de entrada, onde se insere nas doze primeiras linhas da primeira coluna as orientações angulares das camadas e na última linha o valor de  $F$  para o ponto um. Como  $F$  é progressivamente maior, as demais colunas da matriz repetem os valores das 12 primeiras linhas e na última linha insere-se o novo valor de  $F$  para o ponto seguinte, repete-se isso até o ponto 26. Obtem-se assim, uma matriz de entrada  $13 \times 26$ . Para cada coluna da matriz de entrada, há uma saída desejada, o deslocamento transversal correspondente à força aplicada, logo define-se a saída como um vetor  $1 \times 26$ . A Figura 7 apresenta o procedimento para geração de dados para o treinamento da rede neural.

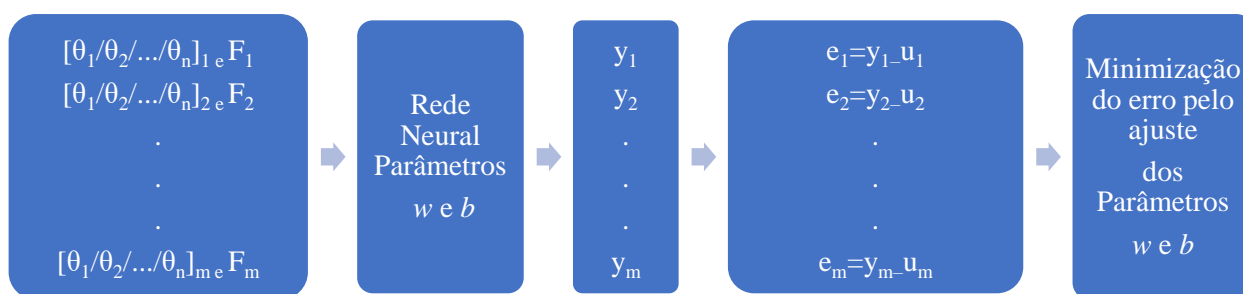
**Figura 7 - Geração de dados para treinamento da rede neural**



**Fonte: Autoria própria**

A rede neural é então treinada, com subsequente verificação de sua capacidade de representar apropriadamente a estrutura proposta. A Figura 8 resume o processo de treinamento da rede neural. Apresentam-se as mesmas entradas que configuraram as diferentes placas compósitas e calcula-se as saídas  $y_j$  ( $j = 1$  a  $20$ ) da rede neural. As saídas da rede são comparadas com as saídas desejadas ( $u$ ) de onde se calcula um erro. Inicia-se então um processo de minimização do erro tendo como variáveis de projeto os parâmetros peso ( $w$ ) e *bias* ( $b$ ) que configuram a rede neural. Quando o erro é satisfatoriamente mínimo, os parâmetros da rede neural encontrados a fazem representar a placa compósita. Isso é checado através da apresentação de novas entradas e comparação com as saídas desejadas.

**Figura 8 - Treinamento da rede neural**



**Fonte: Autoria própria**

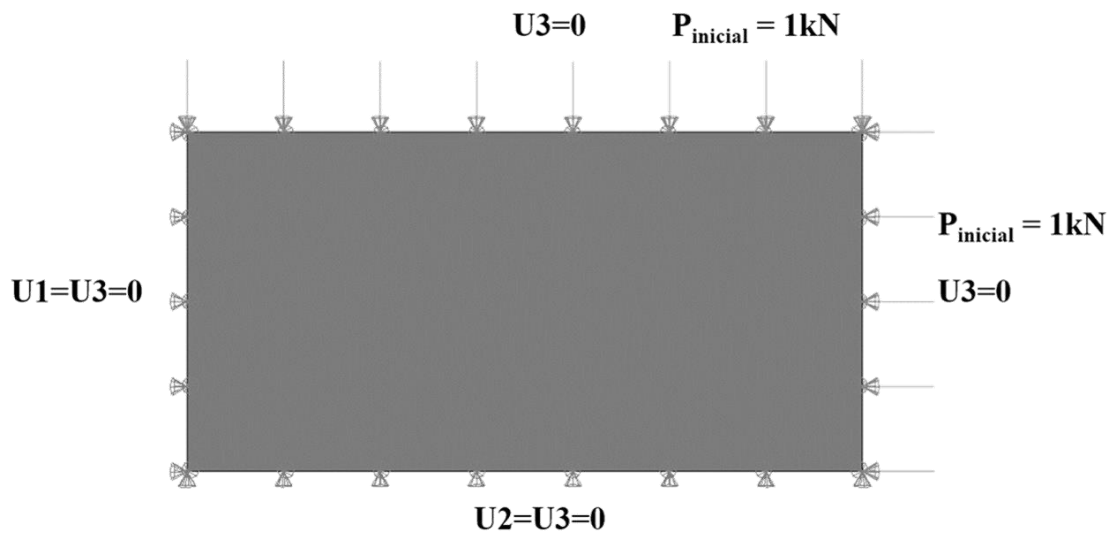
## 4 RESULTADOS

### 4.1 Análise de placas compósitas no Abaqus®

#### 4.1.1 Análise de placas compósitas perfeitas sob carregamento compressivo

Para a presente análise, emprega-se no modelo apresentado em BARBERO (2013), no qual uma placa de orientação  $[(0/90)_3]_s$  é submetida a um carregamento compressivo em duas extremidades. As condições de contorno estão apresentadas na Figura 9.

**Figura 9 - Geometria e condições de contorno da placa analisada.**



**Fonte: Autoria própria**

A placa possui 1000 mm de comprimento e 500 mm de largura. A espessura total da placa é  $Th=10,2$  mm e cada lâmina tem espessura  $t=0,85$  mm.  $U_1$ ,  $U_2$  e  $U_3$  são os deslocamentos nas direções  $x$ ,  $y$  e  $z$ , respectivamente. O material compósito utilizado é a fibra de carbono AS4 com resina epóxi 9310. Os dados do material estão apresentados na Tabela 1.

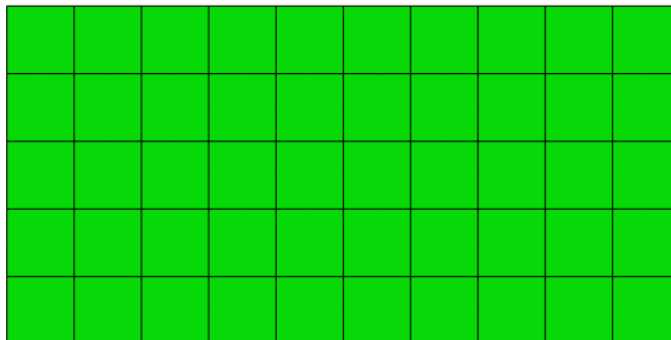
**Tabela 1 - Dados da lâmina de material AS4/9310**

Módulo de Young	Módulo de cisalhamento	Coefficiente de Poisson
$E_1 = 145880$ MPa	$G_{12} = G_{21} = 4386$ MPa	$\nu_{12} = \nu_{13} = 0,263$
$E_2 = E_3 = 13312$ MPa	$G_{23} = 4529$ MPa	$\nu_{23} = 0,470$

**Fonte: (BARBERO, 2013)**

A malha empregada na discretização da placa é apresentada na Figura 10. É composta por 50 elementos S8R5 com cinco graus de liberdade, que é escolhido por se tratar de uma placa fina.

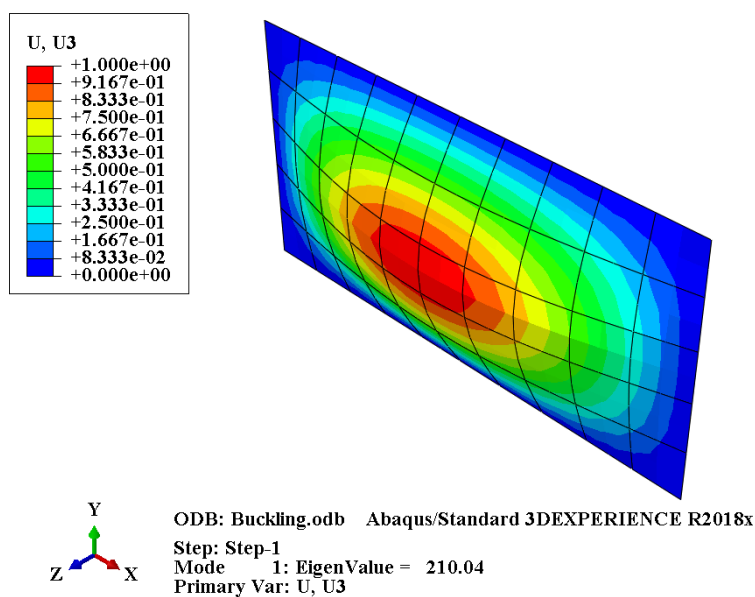
**Figura 10 - Malha de elementos finitos utilizada na discretização da placa.**



**Fonte: Autoria própria**

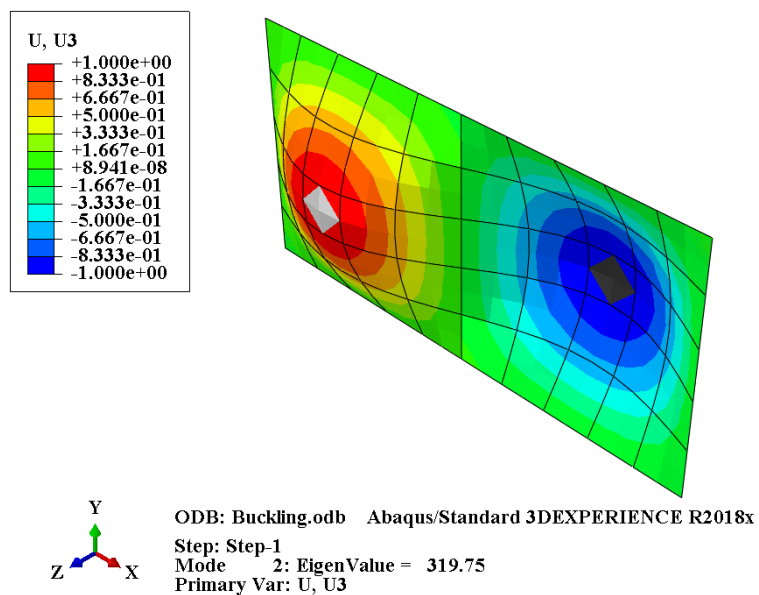
Primeiramente, executa-se a análise para o caso de flambagem linear para os primeiros 4 modos de flambagem. Os valores de carga crítica de flambagem obtidos comparados aos valores fornecidos na literatura estão apresentados na Tabela 2. As Figura 11 à Figura 14 apresentam o aspecto deformado da placa nos quatro primeiros modos de flambagem, sendo U3 a deformação na direção z e dada em milímetros.

**Figura 11 - Primeiro modo de flambagem para o empilhamento [(0/90)<sub>3</sub>]<sub>s</sub>. U3 em milímetros.**



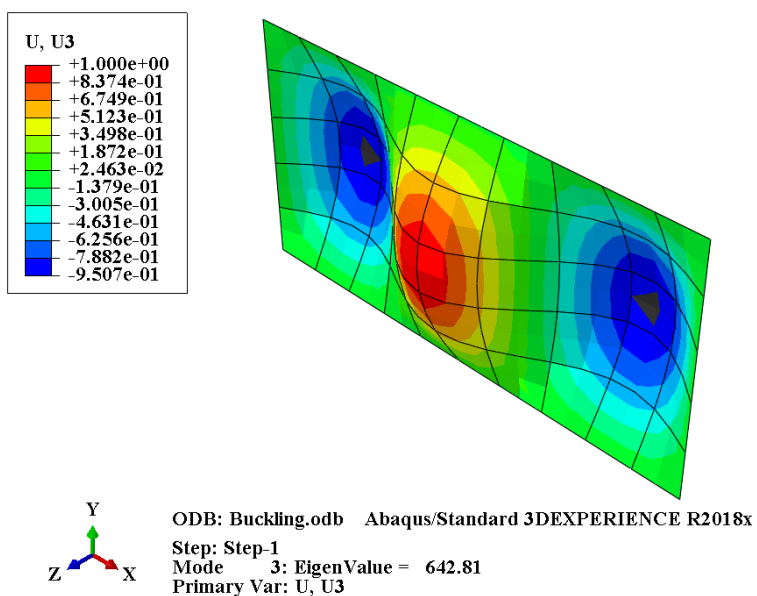
**Fonte: Autoria própria**

Figura 12 - Segundo modo de flambagem para o empilhamento [(0/90)<sub>3</sub>]<sub>s</sub>. U3 em milímetros.



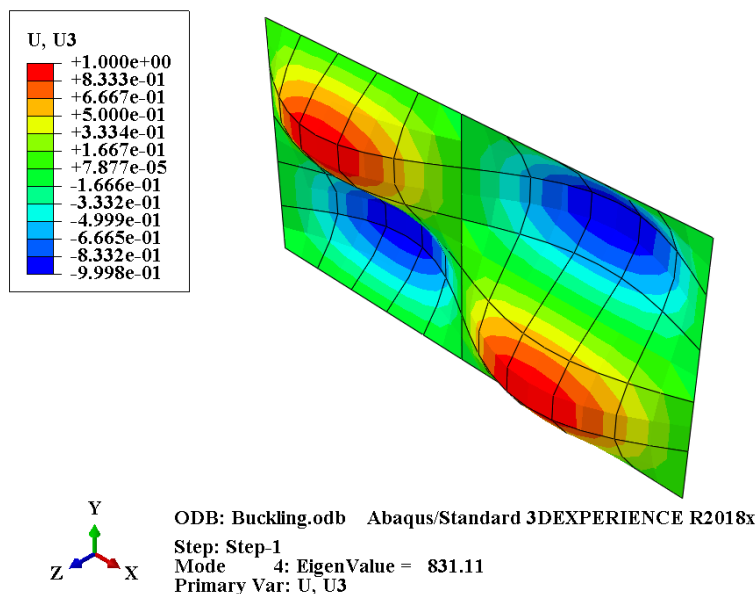
Fonte: Autoria própria

Figura 13 - Terceiro modo de flambagem para o empilhamento [(0/90)<sub>3</sub>]<sub>s</sub>. U3 em milímetros.



Fonte: Autoria própria

Figura 14 - Quarto modo de flambagem para o empilhamento [(0/90)<sub>3</sub>]<sub>s</sub>. U3 em milímetros.



Fonte: Autoria própria

Tabela 2 - Valores de carga de bifurcação para o empilhamento [(0/90)<sub>3</sub>]<sub>s</sub> em N/mm.

Modo de flambagem	(BARBERO, 2013)	Presente trabalho
1	210,10	210,04
2	319,87	319,75
3	643,18	642,81
4	831,65	831,11

Fonte: Autoria própria / (BARBERO, 2013)

Os valores de carga crítica de flambagem obtidos comparados aos valores fornecidos na literatura estão apresentados na Tabela 2.

Os resultados apresentados em BARBERO (2013) são obtidos com o método de entrada de dados do material por matrizes ABDH. Tal método retorna valores cerca de 0,03% a 0,07% diferentes para a carga de bifurcação em comparação aos valores obtidos (Tabela 2). No presente trabalho, opta-se pela utilização do método de entrada de dados do material do tipo LSS (*Laminate Stacking Sequence*). Seleciona-se tal método por proporcionar maior facilidade na modificação das orientações do laminado dentro do Abaqus®.

#### 4.1.2 Análise de placas compósitas imperfeitas sob carregamento compressivo

Após conduzidas as análises lineares, submeteu-se a placa no regime de pós-flambagem através de inserção de imperfeições no material. Faz-se isso com duas premissas diferentes. A primeira com fatores de escala das imperfeições da ordem de 1% da espessura total do laminado ( $Th/100$ ) e a segunda com imperfeições da ordem de 10% da espessura total do laminado ( $Th/10$ ) para o primeiro modo de flambagem.

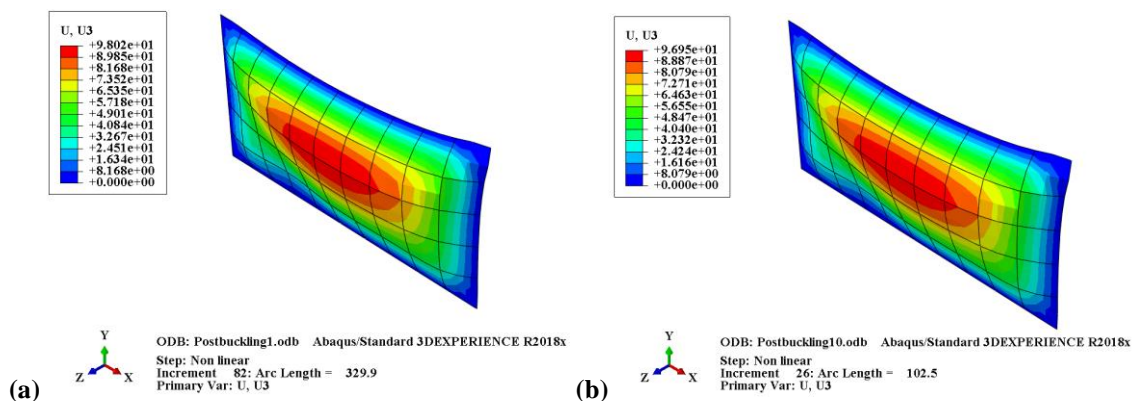
A Tabela 3 apresenta os fatores de escala empregados para os respectivos modos de flambagem e grau de imperfeição. A Figura 15 apresenta o aspecto da placa  $[(0/90)_3]_s$  deformada na direção Z ( $U_3$ ) no regime de pós-flambagem para tais imperfeições.

**Tabela 3 - Fatores de escala utilizados para os respectivos modos de flambagem.**

Modo de flambagem	10% de imperfeição	1% de imperfeição
1	1,02	0,102
2	0,51	0,051
3	0,255	0,0255
4	0,1275	0,01275

Fonte: Autoria própria

**Figura 15 - Aspecto da placa  $[(0/90)_3]_s$  deformada na direção Z ( $U_3$ ) no regime de pós-flambagem para (a) 1% de imperfeição e (b) 10% de imperfeição.  $U_3$  em milímetros.**



Fonte: Autoria própria

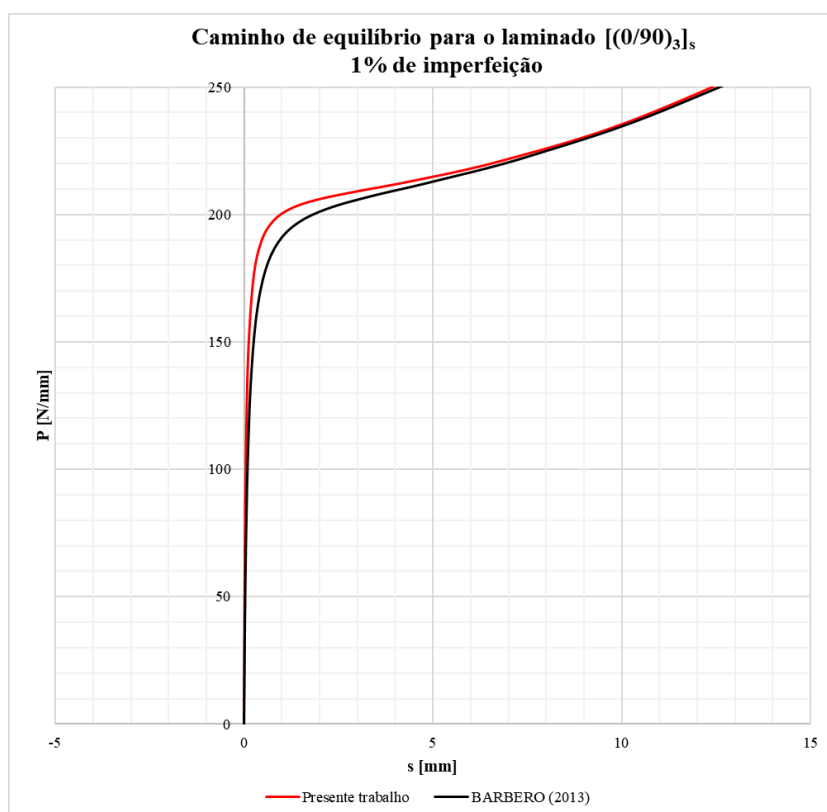
A validação do modelo foi realizada através do comparação entre os resultados obtidos e os exibidos em BARBERO (2013), conforme apresentado nas Figura 16 Figura 17. Ambos os



casos de imperfeição analisados apresentam boa concordância com o modelo apresentado na literatura.

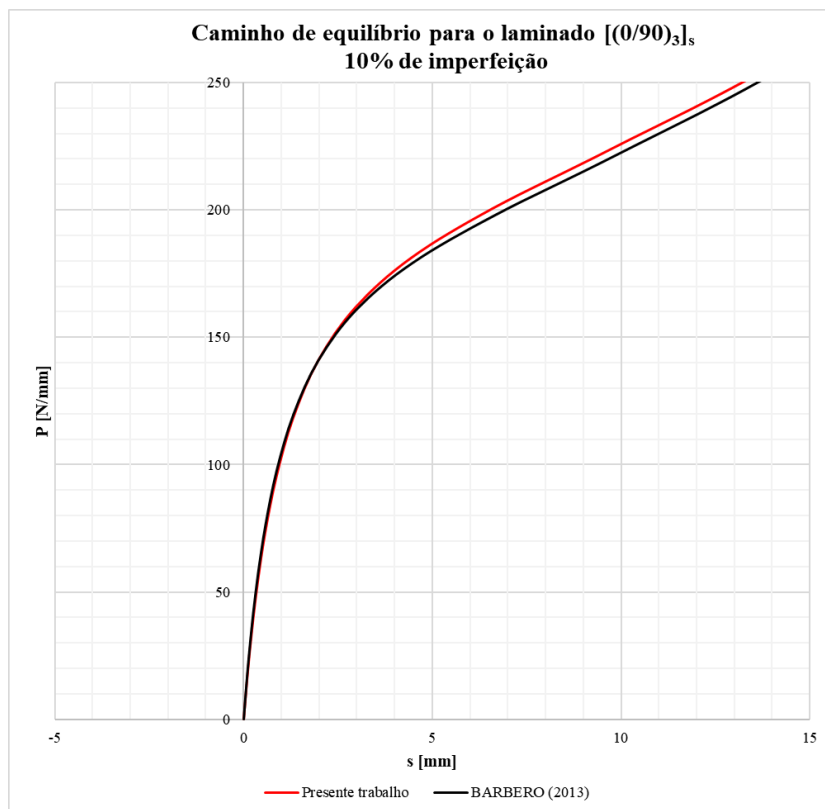
Como mencionado anteriormente, BARBERO (2013) utiliza o método de entrada de dados por matrizes ABDH no Abaqus®, o que retorna valores ligeiramente diferentes para carga de bifurcação (Tabela 2). Uma vez que os modos de flambagem do caso linear são utilizados como entrada para a simulação de pós-flambagem, a diferença no método de entrada dos dados pode explicar a ligeira diferença entre as curvas obtidas comparadas ao modelo de BARBERO (2013).

**Figura 16 - Validação do modelo para o laminado [(0/90)<sub>3</sub>]<sub>s</sub> com 1% de imperfeição.**



**Fonte: Autoria própria**

**Figura 17 - Validação do modelo para o laminado [(0/90)<sub>3</sub>]<sub>s</sub> com 10% de imperfeição.**



**Fonte: Autoria própria**

## 4.2 Definição e modelagem de 20 placas com diferentes orientações angulares

Seguindo a metodologia definida, opta-se por modelar e coletar os resultados (Abaqus®) de vinte placas com orientações angulares distintas. Resultados estes que servem de entrada na etapa de aproximação por redes neurais. Logo, a fim de definir os empilhamentos que melhor representam o campo amostral, é utilizado o método de DoE (*Design of Experiments*) e realiza-se um planejamento fatorial com número de fatores igual a seis.

O laminado é composto por doze camadas, divididas simetricamente. Sendo assim, define-se seis opções angulares distintas para cada uma das camadas, podendo se repetir no laminado. As orientações empregadas são:  $0^\circ$ ,  $\pm 30^\circ$ ,  $\pm 45^\circ$  ou  $90^\circ$ .

As combinações de entrada no DoE, apresentadas acima, fornecem 46656 empilhamentos diferentes. A ferramenta retorna as vinte configurações angulares que melhor representam o campo amostral dado, Tabela 3. A coluna “Ordem” define qual a posição da configuração angular dentro das 46656 combinações angulares possíveis.

**Tabela 4 - Configurações angulares obtidas através do DoE**

Placa	Ordem	Empilhamento
1	46278	[90/90/45/-30/0/90]s
2	42776	(90 /30/-45/-45/-30/-30)s
3	16795	[0/-45/90/45/30/-45]s
4	34684	[45/0/45/30/0/30]s
5	23873	[30/-45/0/30/-45/45]s
6	13191	[-30/45/-30/-45/0/0]s
7	6810	[-45/90/-30/30/-45/90]s
8	21742	[0/45/45/30/90/30]s
9	30255	[30/90/0/-45/0/0]s
10	30324	[30/90/0/0/-30/90]s
11	5158	[-45/30/90/90/-30/30]s
12	11569	[-30/0/90/30/0/-45]s
13	38330	[45/90/30/0/45/-30]s
14	38849	[45/90/90/90/-45/45]s
15	13940	[-30/45/45/30/-30/-30]s
16	22631	[0/90/0/45/30/45]s
17	5947	[-45/45/30/30/-30/-30]s
18	43179	[90/30/-30/90/0/0]s
19	20475	[0/30/45/45/45/0]s
20	40450	[90/-30/-30/-30/30/30]s

**Fonte: Autoria própria**

### 4.3 Resultados Redes Neurais

Com os resultados das 20 placas obtidos através do Abaqus®, inicia-se a metamodelagem. Para isso utiliza-se a *toolbox Neural Fitting Tool (Nftool)* disponível no MATLAB®.

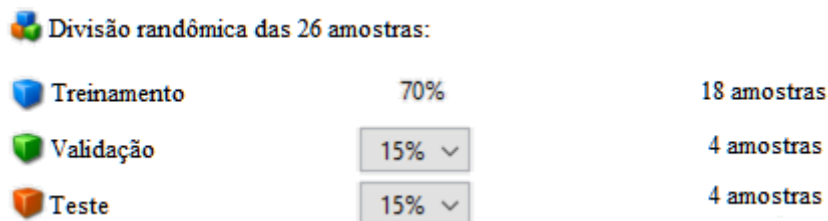
Primeiramente, são definidas duas matrizes (entrada e saída) a serem utilizadas no processo de treinamento da rede neural. A entrada do treinamento da rede neural é uma matriz 13x26. Sendo as primeiras 12 linhas da matriz a sequência de empilhamento da placa, que se repetem em todas as 26 colunas, e a última linha as cargas compressivas, dadas em Newton por milímetro, para cada um dos 26 pontos obtidos no Abaqus®. A saída almejada pela rede é um vetor 1x26, representando os deslocamentos transversais correspondentes às cargas de entrada, na direção Z em milímetros.

Para realizar a metamodelagem, a ferramenta do MATLAB® divide os dados de entrada e saída em 3 subgrupos:

- Treinamento: Parte do campo amostral apresentada à rede para treinamento da mesma e, é ajustada de acordo com o erro;

- Validação: Parte do campo amostral utilizada para medir a generalização da rede. Tem a função de parar o treinamento quando a generalização para de apresentar melhoras;
- Teste: Parte do campo amostral que não produz efeito nenhum ao treinamento da rede, fornece uma medida independente da performance da rede neural e é utilizada em análises pós treinamento.

Figura 18 - Definição da amostra da rede

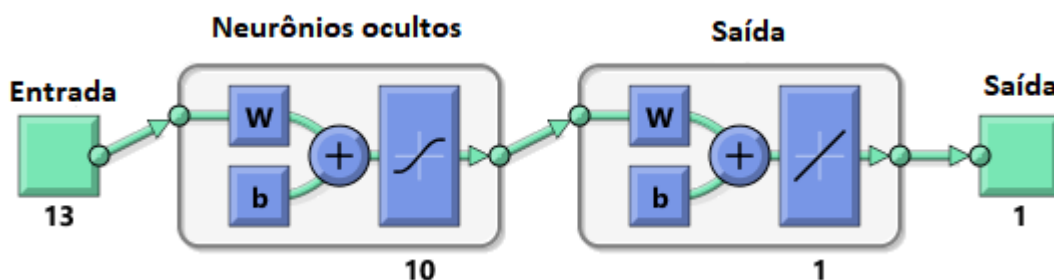


Fonte: Autoria própria

Com o intuito de otimizar o processo de aproximação dos resultados da rede, a ferramenta possibilita que os dados sejam divididos entre três subgrupos acima. Desta forma, é possível adequar a rede de acordo com as necessidades e parâmetros de cada análise.

No presente trabalho, opta-se pela divisão do campo amostral apresentada na Figura 18. Além disso, a ferramenta de redes neurais permite a alteração do número de neurônios presentes na camada oculta da rede e o número de camadas ocultas da rede. Para o presente estudo, emprega-se uma camada oculta com 10 neurônios.

Figura 19 - Número de neurônios na camada oculta da rede



Fonte: Autoria própria

A análise consiste em escolher uma das 20 placas modeladas via Abaqus® e treinar uma rede neural, que através da sua entrada (configuração angular e cargas) aproxime as saídas reais obtidas (deformações). Em seguida, a rede neural treinada para esta placa é testada utilizando entradas (configuração angular e cargas) referentes a placas diferentes, e então a nova saída (deformações) obtida através do metamodelo testado é comparada com a saída obtida através do Abaqus®. Isto é feito a fim de avaliar a capacidade da rede neural treinada para uma placa em aproximar os resultados reais de outras placas.

Inicialmente, traça-se a sobreposição dos gráficos de carga compressiva *versus* deformação, obtidos para as vinte placas analisadas no Abaqus®, com 1% e 10% de imperfeição. Então, define-se qual modelo é empregado no treinamento, e em seguida testa-se a rede neural para outras placas selecionadas, com entradas e saídas distintas.

A fim de verificar a capacidade de aproximação da resposta (deformações) da rede neural, em comparação com a resposta obtida via análise de elementos finitos, utiliza-se a ferramenta estatística de regressão linear. Que consiste em usar uma equação para expressar a relação entre as variáveis de interesse (resposta) e o conjunto de variáveis relacionadas esperadas. Na maioria dos casos, a equação de regressão é apenas uma aproximação da verdadeira relação funcional entre as variáveis de interesse.

$$y = \beta_0 + \beta_1x_1 + \beta_2x_2 + \dots + \beta_kx_k + \varepsilon \quad (9)$$

A Equação 9 representa o modelo de regressão  $y$ , em termos das variáveis independentes  $x$ , dos coeficientes de regressão  $\beta$  e do erro  $\varepsilon$ .

A análise de regressão depende da determinação dos parâmetros desconhecidos  $\beta$ , este processo é chamado de adequação do modelo aos dados.

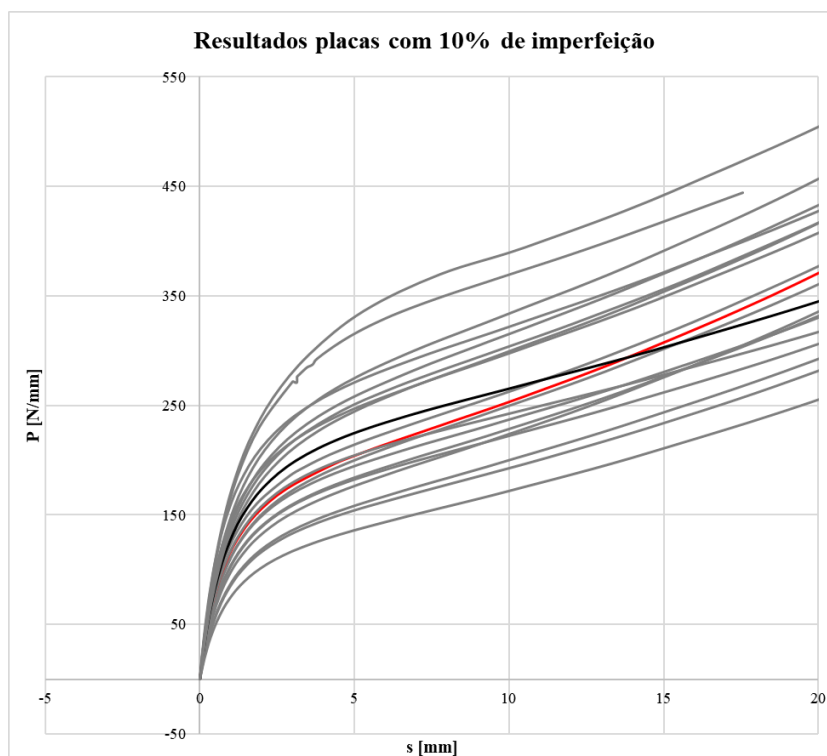
O próximo passo da análise é chamado de checagem de adequação do modelo, onde a apropriação do modelo é estudada e a qualidade da adequação é acertada. Através desta análise a utilidade do modelo de regressão pode ser determinada (MONTGOMERY, et al., 2012).

Para o presente estudo, utiliza-se a função *plotregression* do MATLAB®, a qual é recomendada pela ferramenta de redes neurais *Nftools* para análise da aproximação de suas respostas.

#### **4.3.1 Comparação entre a placa 10 e a placa 17 com imperfeição de 10%**

Através do gráfico que compara o comportamento dos 20 modelos obtidos no Abaqus® com 10% de imperfeição, opta-se por treinar uma rede neural com os resultados da placa de número 10. Pois ela apresenta um comportamento que se aproxima de um comportamento médio das outras placas.

**Figura 20- Resultados das análises no Abaqus® para placas com 10% de imperfeição. Placas 10 e 17 apresentadas em vermelho e preto respectivamente**



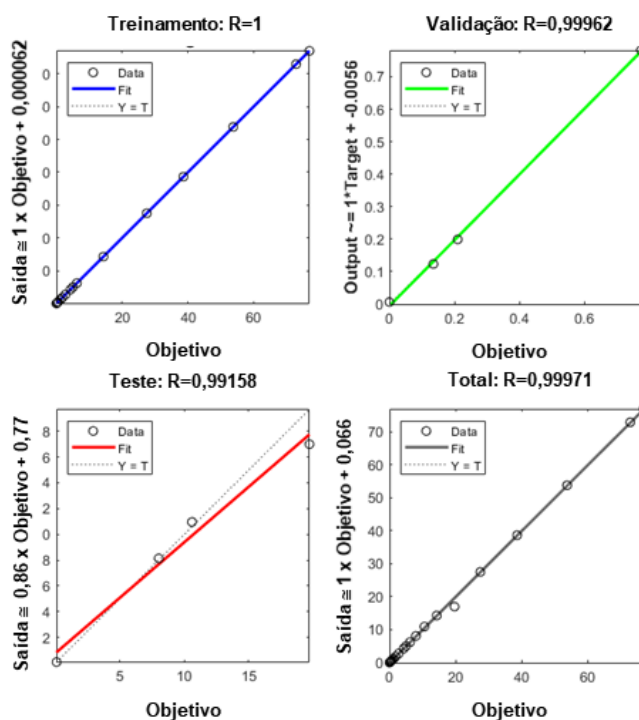
**Fonte: Autoria própria**

Treina-se uma rede com as entradas (cargas e orientação angular) e saídas (deformações) da placa 10. Retreina-se novamente até atingir uma regressão ótima, ou seja, com um erro aceitável entre a saída obtida na rede neural e a saída obtida no Abaqus®.

Em seguida, testa-se a rede para a placa 17. Para isso, a rede neural agora passa a ser alimentada com as entradas da placa 17, mas suas deformações (saídas) não são mais fornecidas à rede neural.

Optou-se por comparar as duas placas em questão pois apresentam comportamentos similares, como pode ser observado na Figura 20.

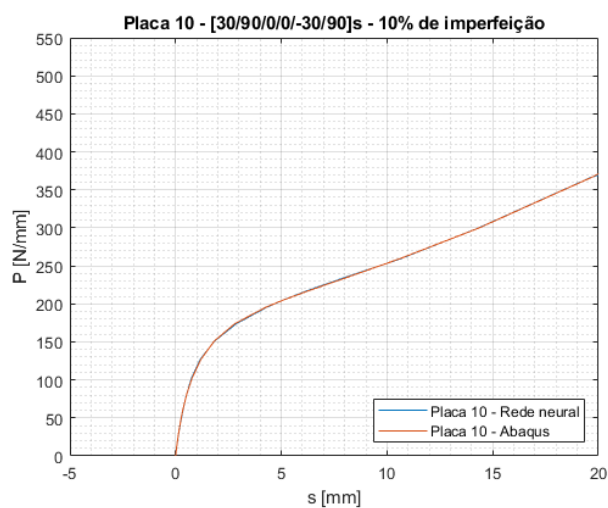
Figura 21 - Análise de regressão da rede neural treinada para a placa 10 com 10% de imperfeição



Fonte: Autoria própria

A Figura 21 mede o erro da aproximação dos dados de saída da rede neural (deformação) com as deformações obtidas via Abaqus® para a placa 10 com 10% de imperfeição.

Figura 22 - Comparação da saída da rede neural com os resultados obtidos via Abaqus® para a placa 10

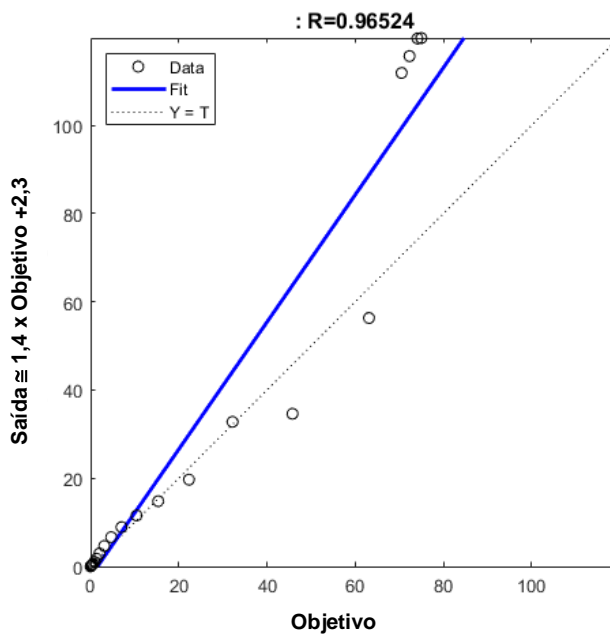


Fonte: Autoria própria

Pode-se observar que a rede neural treinada para a placa 10 apresenta resultados satisfatórios. Como mostra a Figura 22, ela possui uma boa capacidade de representar os resultados obtidos via Abaqus®.

A fim de testar a capacidade da rede neural, em fornecer saídas para entradas desconhecidas, aplica-se as entradas (cargas e orientação angular) da placa 17.

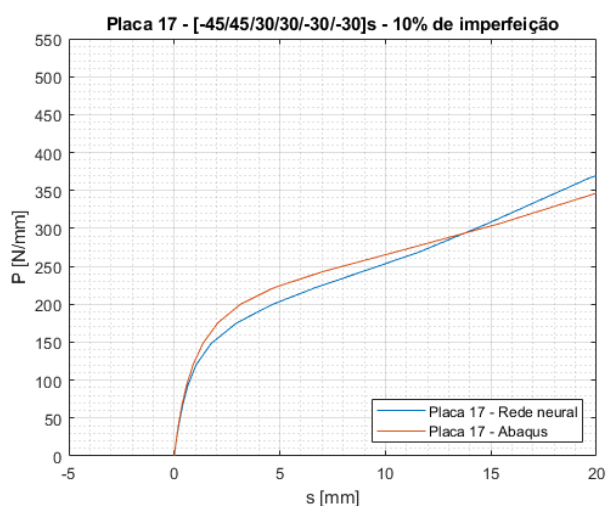
**Figura 23 - Análise de regressão da rede neural testada para a placa 17 com 10% de imperfeição**



**Fonte: Autoria própria**

A Figura 23 mede o erro da aproximação dos resultados obtidos pela rede neural, treinada para a placa 10 e testada com as entradas da placa 17.

**Figura 24 - Comparação dos resultados da rede neural testada para a placa 17 com resultados obtidos via Abaqus®**



**Fonte: Autoria própria**

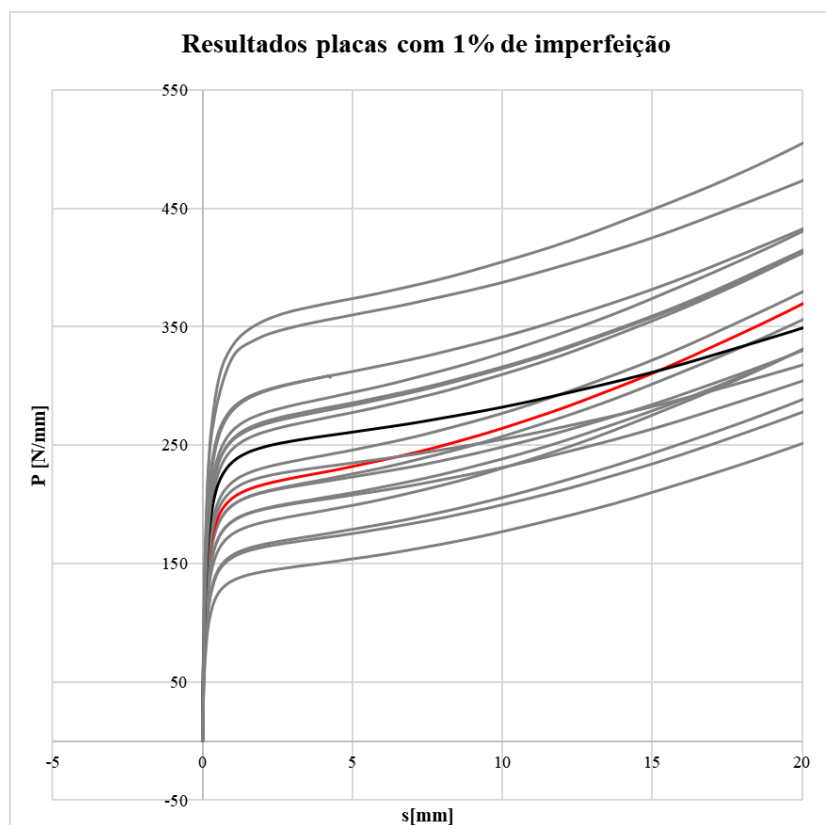
Observa-se, vide Figura 24, que a rede neural treinada para a placa 10, ao ser alimentada com as entradas da placa 17, retorna saídas que apresentam uma boa aproximação com os dados obtidos via *software* de elementos finitos.



### 4.3.2 Comparação entre a placa 10 e a placa 17 com imperfeição de 1%

A fim de obter mais elementos de análise da eficiência da aproximação da rede neural, repete-se o mesmo procedimento realizado no item anterior, mas agora com os dados obtidos através de modelagem de placas com 1% de imperfeição.

**Figura 25 - Resultados das análises no Abaqus® para placas com 1% de imperfeição. Placas 10 e 17 apresentadas em vermelho e preto respectivamente**

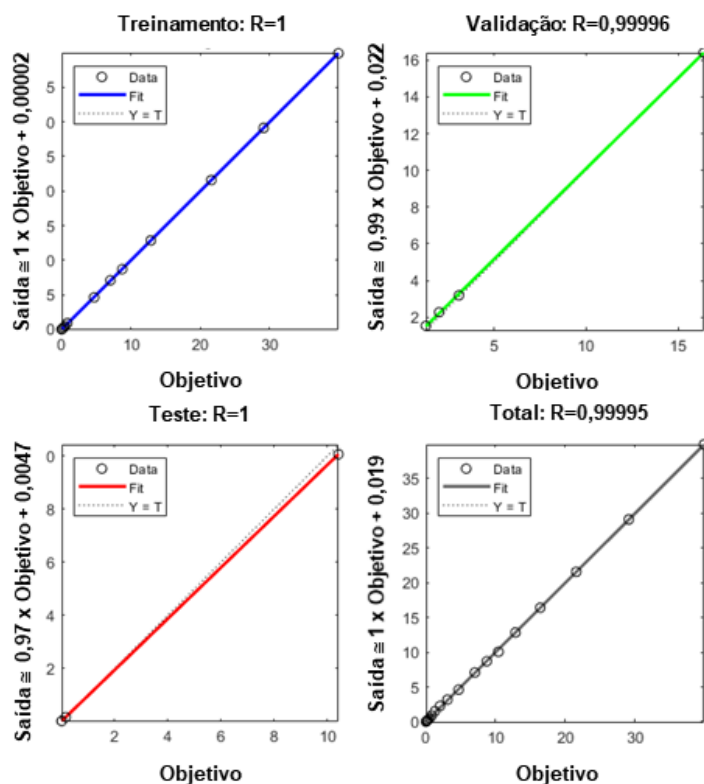


**Fonte: Autoria própria**

Treina-se, novamente, uma rede com as entradas e saídas da placa 10, mas agora para uma placa modelada com 1% de imperfeição. Retreina-se a rede até atingir uma regressão ótima. Em seguida, testa-se a rede para a placa 17.

Ou seja, a rede neural agora passa a ser alimentada com as entradas (cargas e orientações angulares) da placa 17 com 1% de imperfeição, mas não recebe mais suas saídas (deformações).

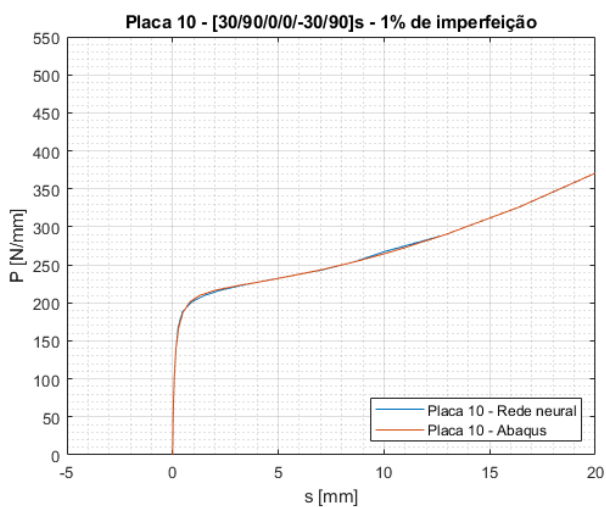
Figura 26- Análise de regressão da rede neural treinada para a placa 10 com 1% de imperfeição



Fonte: Autoria própria

A Figura 26 mede o erro da aproximação dos dados de saída da rede neural (deformação) com as deformações obtidas via Abaqus® para a placa 10 com 1% de imperfeição.

Figura 27 - Comparação da saída da rede neural com os resultados obtidos via Abaqus® para a placa 10

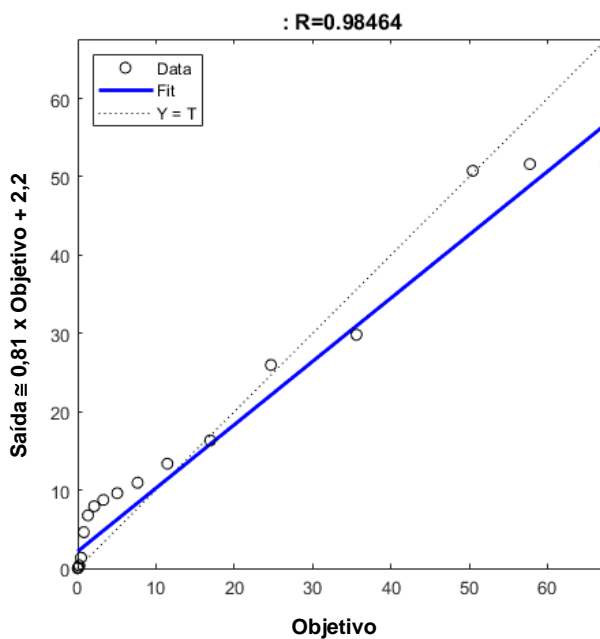


Fonte: Autoria própria

Assim como é observado para a placa modelada com 10% de imperfeição, nota-se que a rede neural treinada para a placa 10 com 1% também apresenta resultados satisfatórios. Ou seja, ela possui uma boa capacidade de aproximação dos resultados obtidos via Abaqus®.

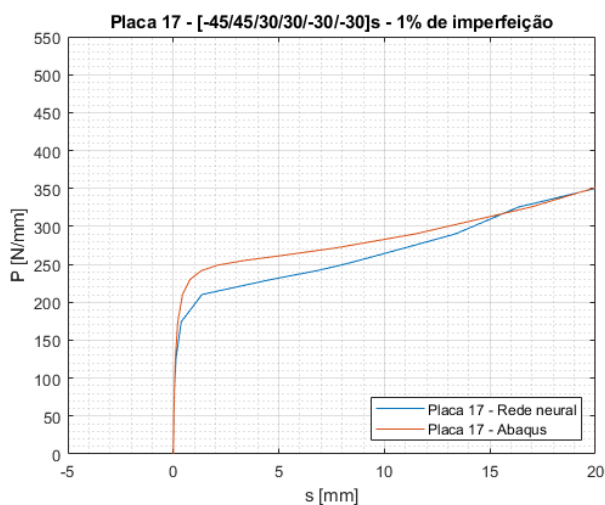
Em seguida, repete-se o procedimento de testar a rede neural treinada com entradas referentes a placa 17 modelada com 1% de imperfeição.

**Figura 28 - Análise de regressão da rede neural testada para a placa 17 com 1% de imperfeição**



Fonte: Autoria própria

**Figura 29 - Comparação dos resultados da rede neural testada para a placa 17 com resultados obtidos via Abaqus®**



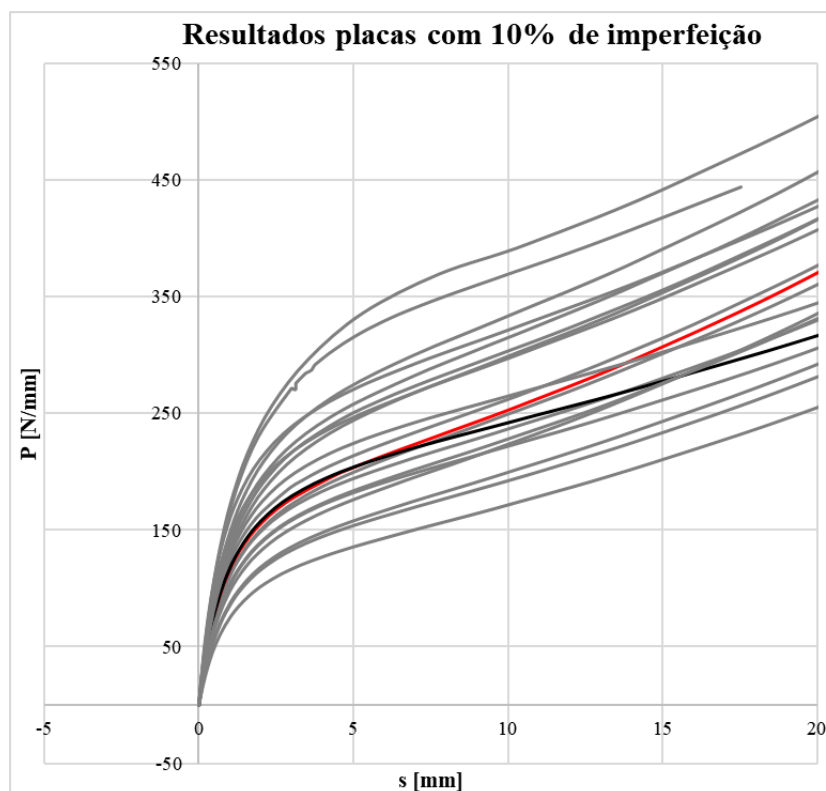
Fonte: Autoria própria

Nota-se, através da Figura 29, que a rede neural treinada para a placa 10, ao ser alimentada com as entradas (cargas e orientações angulares) da placa 17 modelada com 1% de imperfeição, também retorna saídas (deformações) que apresentam uma boa aproximação aos dados obtidos pelo Abaqus®.

#### 4.3.3 Comparação entre a placa 10 e a placa 15 com imperfeição de 10%

A fim de continuar analisando a capacidade de aproximação, testa-se a mesma rede neural treinada anteriormente (placa 10 com 10% de imperfeição), com entradas de uma placa com comportamento ligeiramente diferente aos da placa 17.

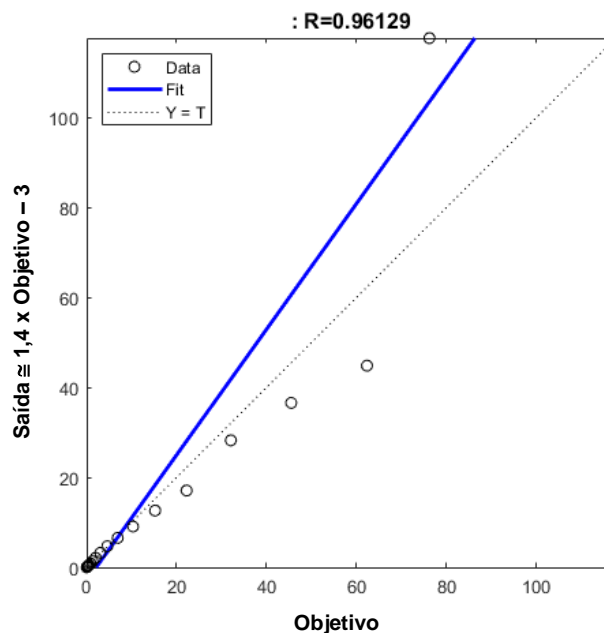
**Figura 30 - Resultados das análises no Abaqus® para placas com 10% de imperfeição. Placas 10 e 15 apresentadas em vermelho e preto respectivamente**



Fonte: Autoria própria

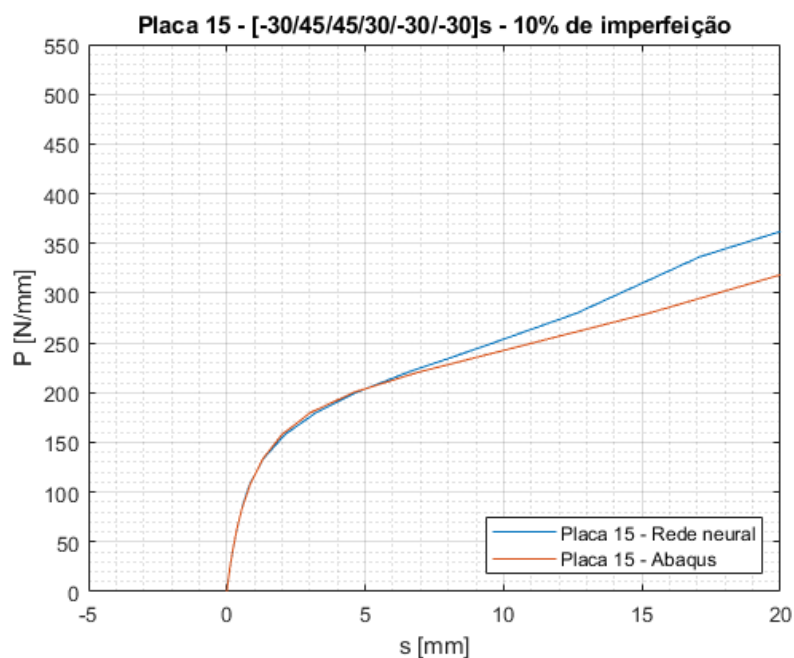
Através da Figura 30 - , optou-se por testar a rede neural com as entradas da placa 15 modelada com 10 % de imperfeição.

Figura 31 - Análise de regressão da rede neural testada para a placa 15 com 10% de imperfeição



Fonte: Autoria própria

Figura 32 - Comparação dos resultados da rede neural testada para a placa 15 com resultados obtidos via Abaqus®



Fonte: Autoria própria

Observa-se, vide

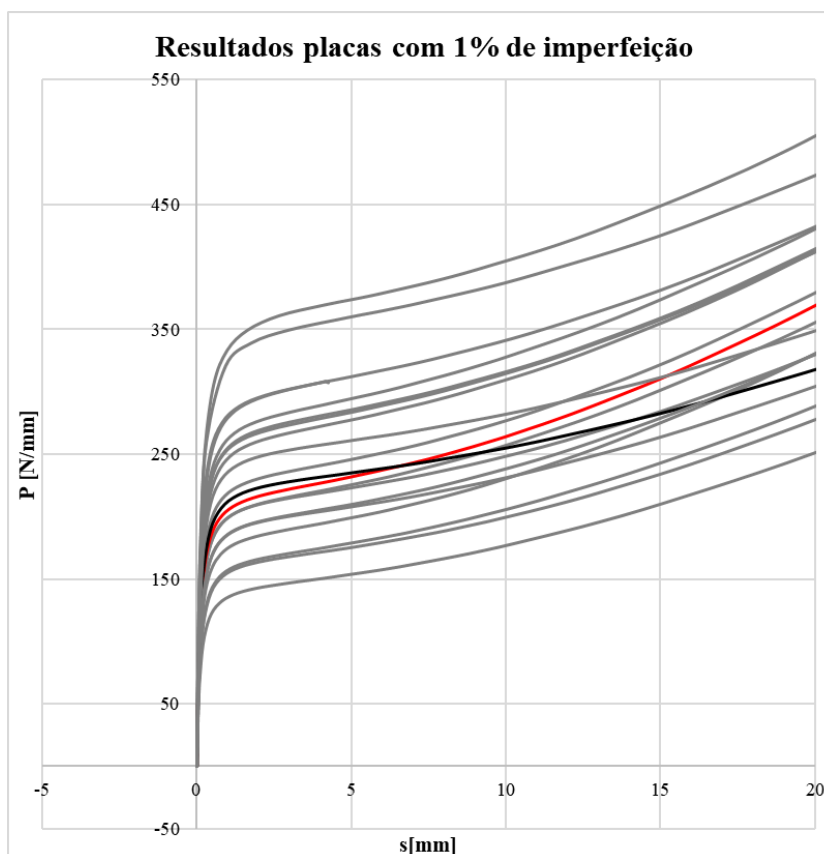
Figura 32, que a rede neural treinada para a placa 10 com 10% de imperfeição, ao ser testada com as entradas (cargas e orientações angulares) da placa 15, retorna saídas

(deformações) que ainda apresentam uma aproximação razoável com os dados obtidos via Abaqus®. Nota-se, através da análise de regressão, que os resultados apresentados pela rede neural, treinada para a placa 10 com 1% de imperfeição, e testada com as entradas de duas placas diferentes, são equivalentes,  $R=0,96129$  para a placa 15 e  $R=0,96524$  para a placa 17.

#### 4.3.4 Comparação entre a placa 10 e a placa 15 com imperfeição de 1%

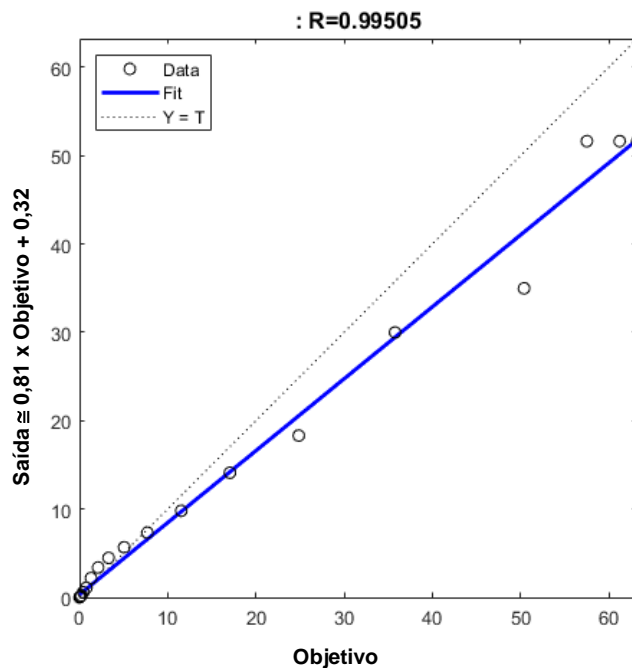
Em seguida, repete-se o procedimento de teste mais uma vez. Agora testando a rede neural treinada para a placa 10 com 1% de imperfeição, com as entradas da placa 15 modelada com 1% de imperfeição.

**Figura 33 - Resultados das análises no Abaqus® para placas com 1% de imperfeição. Placas 10 e 15 apresentadas em vermelho e preto respectivamente**



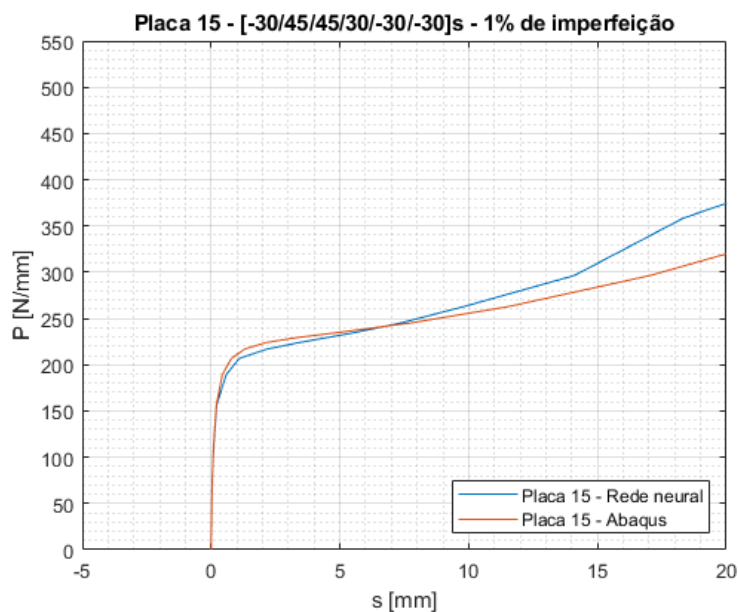
Fonte: Autoria própria

Figura 34 - Análise de regressão da rede neural testada para a placa 15 com 1% de imperfeição



Fonte: Autoria própria

Figura 35 - Comparação dos resultados da rede neural testada para a placa 15 com resultados obtidos via Abaqus®



Fonte: Autoria própria

Observa-se, vide

Figura 35, que a rede neural treinada para a placa 10 com 1% de imperfeição, ao ser testada com as entradas da placa 15 com 1% de imperfeição, também retorna saídas com uma

aproximação razoável em relação aos dados obtidos via elementos finitos. Através da análise de regressão, notou-se uma melhora na aproximação ( $R=0,99505$ ) para placa 15 com 1% de imperfeição, se comparada a aproximação obtida para a placa 17 com 1% de imperfeição ( $R=0,9846$ ).

#### 4.3.5 Comparação da placa 10 com as placas 5, 8, 4 e 19, com 1% de imperfeição

Observa-se, da Tabela 5, que a rede neural tem um desempenho melhor com as placas modeladas com 1% de imperfeição.

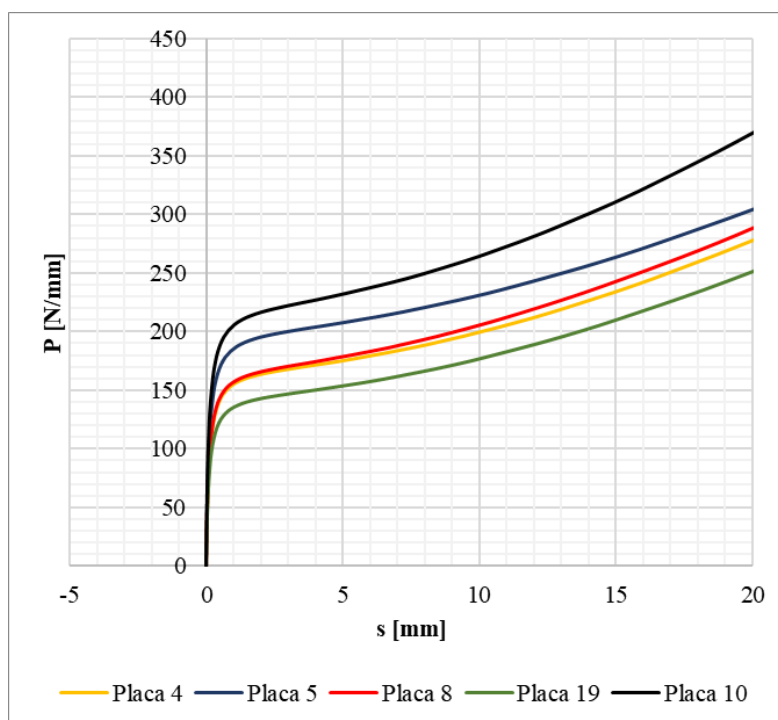
**Tabela 5 - Comparação da análise de regressão (R)**

Imperfeição (%)	Placa 15	Placa 17
10	0,96129	0,96524
1	0,99505	0,98464

**Fonte: Autoria própria**

Com o intuito de continuar testando a capacidade de aproximação da rede neural, repete-se os procedimentos já realizados. Testou-se a rede neural treinada para a placa 10 com 1% de imperfeição, pois apresentaram um resultado melhor, e utiliza-se entradas (cargas e orientações angulares) de placas diferentes. Para isso, escolhe-se analisar placas com curvas cada vez mais afastadas da curva 10. Placas 5, 8, 4 e 19 (pior caso de análise).

**Figura 36 - Comparação entre placas modeladas com 1% de imperfeição**



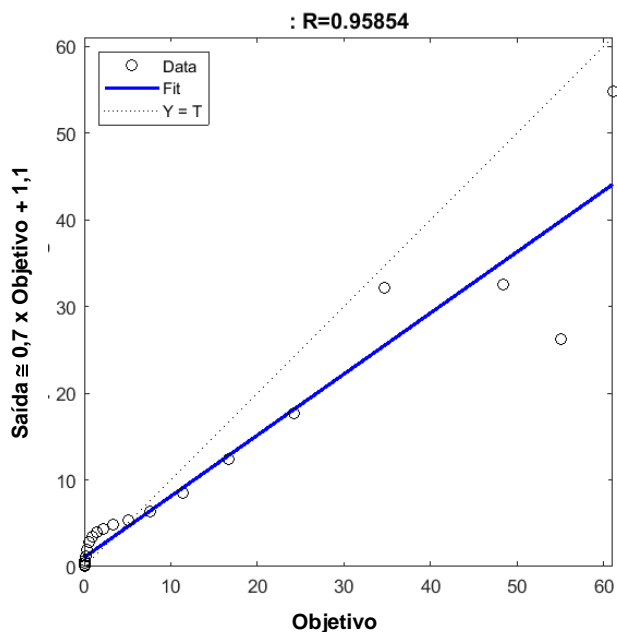
**Fonte: Autoria própria**



A Figura 36 ilustra a discrepância do comportamento das placas escolhidas para análise em relação a placa 10, a qual fornece os dados de entrada para o treinamento da rede neural.

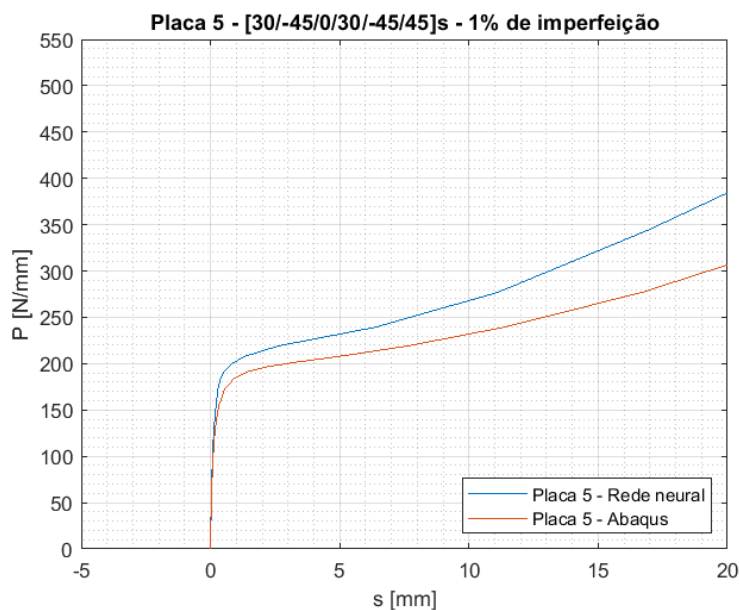
Efetuada a análise através da *toolbox* do MATLAB®, obtem-se os seguintes resultados:

**Figura 37 - Análise de regressão da rede neural testada para a placa 5 com 1% de imperfeição**



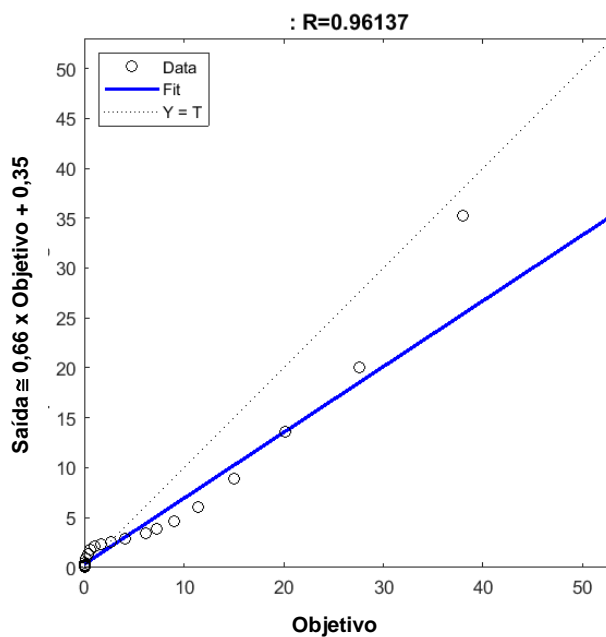
Fonte: Autoria própria

**Figura 38 - Comparação dos resultados da rede neural testada para a placa 5 com resultados obtidos via Abaqus®**



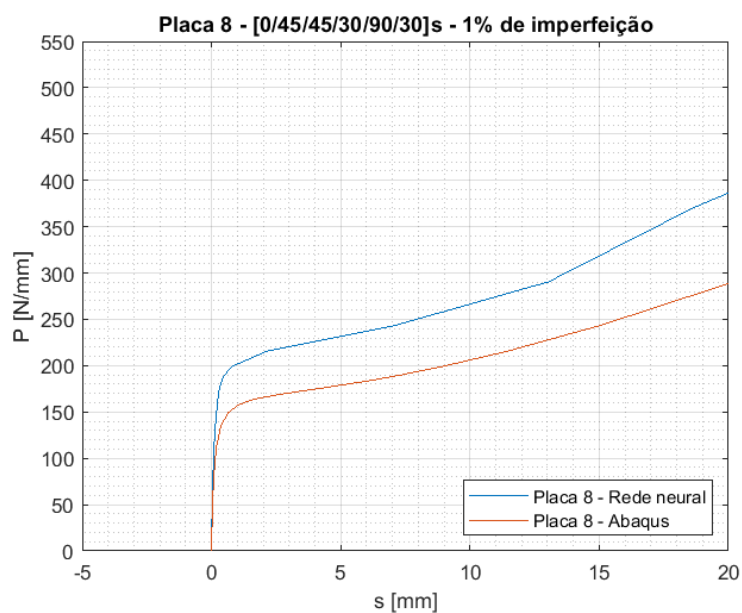
Fonte: Autoria própria

Figura 39 - Análise de regressão da rede neural testada para a placa 8 com 1% de imperfeição



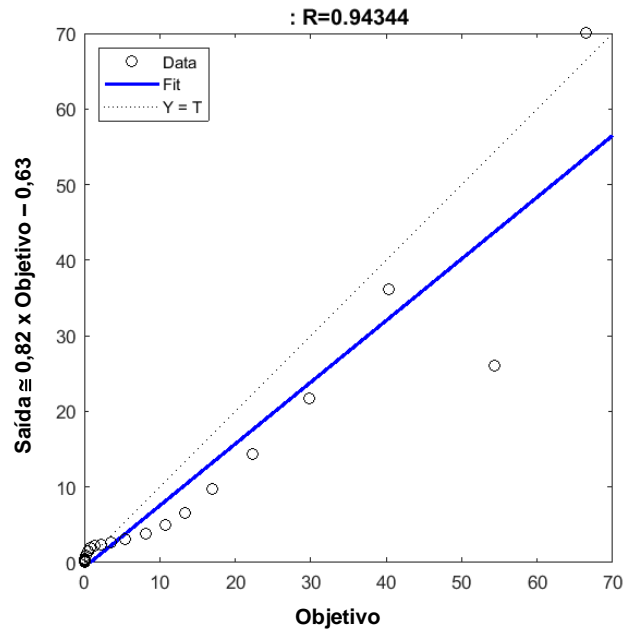
Fonte: Autoria própria

Figura 40 - Comparação dos resultados da rede neural testada para a placa 8 com resultados obtidos via Abaqus®



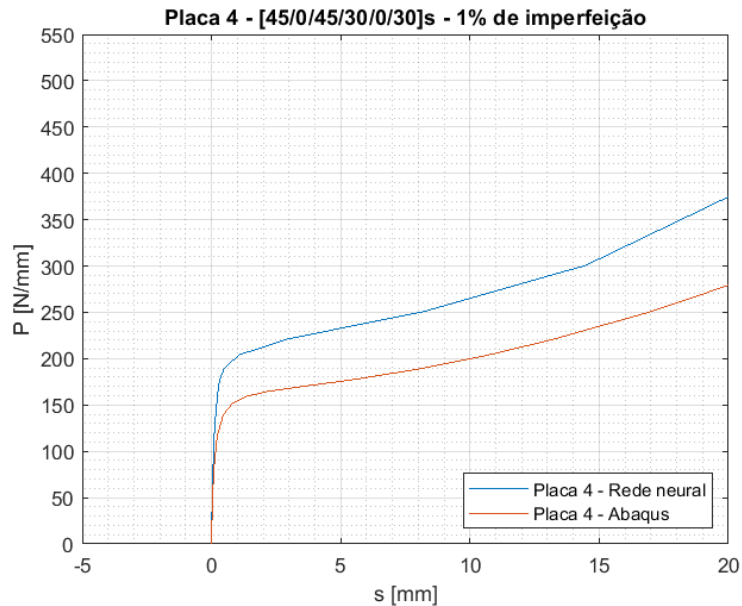
Fonte: Autoria própria

Figura 41 - Análise de regressão da rede neural testada para a placa 4 com 1% de imperfeição



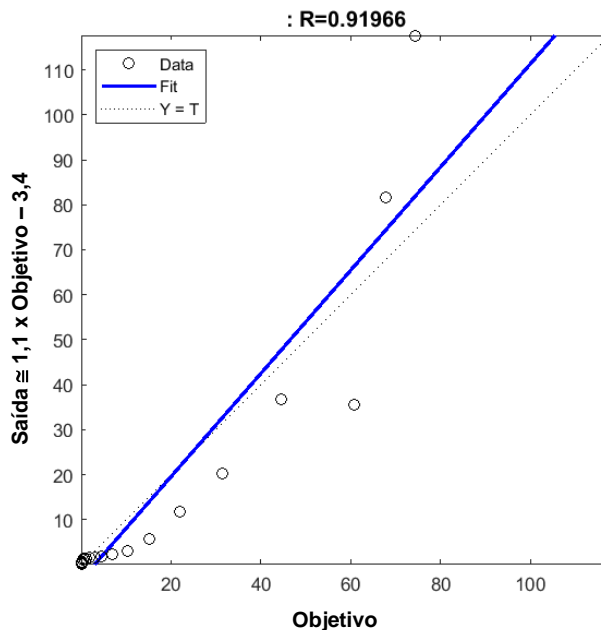
Fonte: Autoria própria

Figura 42 - Comparação dos resultados da rede neural testada para a placa 4 com resultados obtidos via Abaqus®



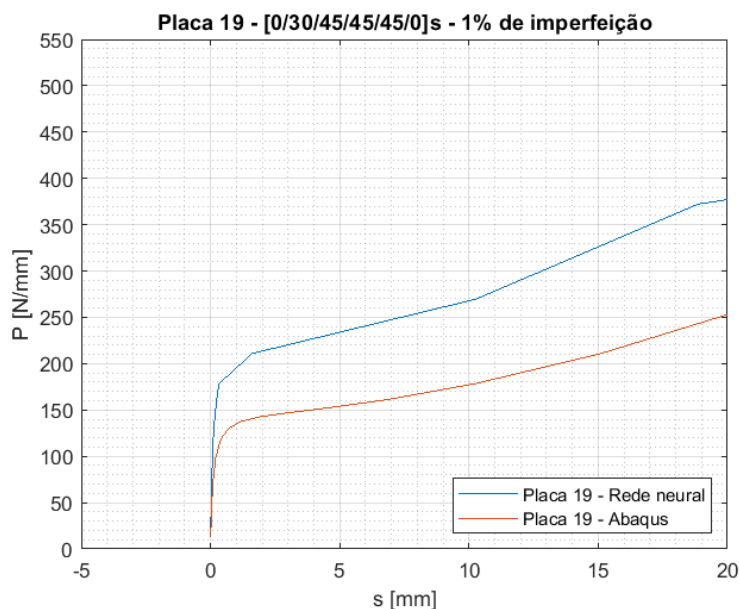
Fonte: Autoria própria

**Figura 43 - Análise de regressão da rede neural testada para a placa 19 com 1% de imperfeição**



**Fonte: Autoria própria**

**Figura 44 - Comparação dos resultados da rede neural testada para a placa 19 com resultados obtidos via Abaqus®**



**Fonte: Autoria própria**

Desta forma, nota-se que uma rede neural treinada para uma placa possui dificuldade em aproximar resultados de placas com comportamentos muito discrepantes daquela a qual realizou o treinamento da rede.

#### 4.3.6 Modificação dos parâmetros da *toolbox* de redes neurais.

Com o intuito de melhorar o desempenho da rede neural, opta-se por treinar a rede com as entradas (cargas e orientações angulares) da placa 10 com 1% de imperfeição e testar para as entradas da placa 19, a qual se mostrou ser o pior cenário de análise. Mas desta vez alterando dois parâmetros no treinamento da rede neural: o número de neurônios da camada oculta e o percentual da amostra destinada à teste. A análise consiste em:

- treinar a rede, com entradas da placa 10, alterando os parâmetros conforme Tabela 5;
- retreinar a rede até obter uma regressão aceitável, R superior à 0,99;
- testar a rede com as entradas da placa 19, a fim de verificar qual combinação de parâmetros retorna a melhor resposta (maior regressão R).

Primeiramente opta-se por aumentar o número de neurônios, mas as aproximações passaram a divergir cada vez mais. Logo, define-se que a análise será realizada a partir da diminuição do número de neurônios.

**Tabela 6 - Regressão (R) do teste das redes treinadas com diferentes configurações de parâmetros da *toolbox***

Número de neurônios \ Percentual da amostra (teste)	20%	25%	30%
9	0,9782	0,9843	0,98378
8	0,88823	0,95547	0,94092
7	0,98414	0,98141	0,98275
6	0,93763	0,91966	0,97084
5	0,98039	0,97584	0,95343
4	0,97278	0,9745	0,98316
3	0,94947	0,98406	0,98407
9	0,9782	0,9843	0,98378
8	0,88823	0,95547	0,94092
7	0,98414	0,98141	0,98275
6	0,93763	0,91966	0,97084

**Fonte: Autoria própria**

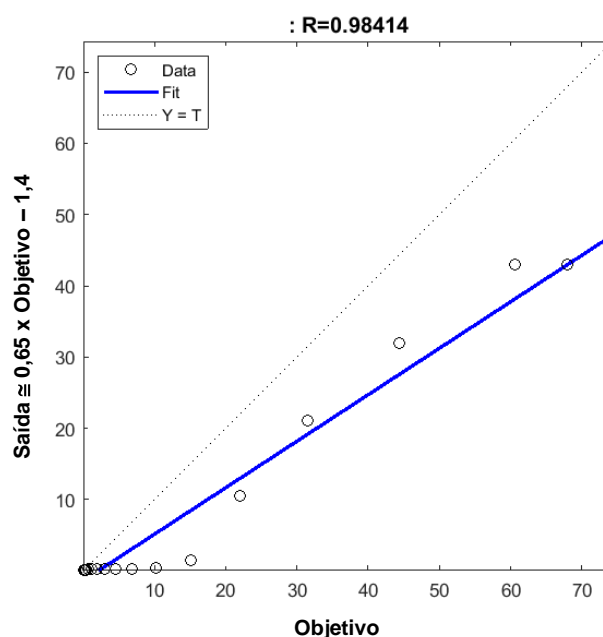
A regressão (R) da Tabela 5 compara as saídas (deformações) da rede neural, treinada com a placa 10 e testada com as entradas da placa 19, com as saídas da placa 19 obtidas via elementos finitos.

Conforme aponta a Tabela 6, realiza-se 21 novas simulações. A tabela tem o objetivo de comparar e determinar com qual configuração uma rede neural retorna a melhor resposta.

Ou seja, a melhor capacidade de aproximação, ao ser testada com entradas referentes a uma placa diferente da qual é utilizada em seu treinamento.

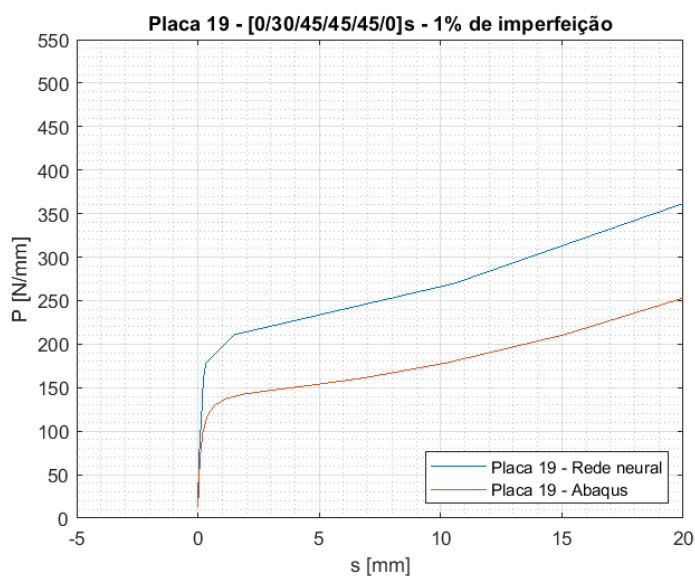
Através desta análise, chega-se a uma configuração ótima de 7 neurônios na camada oculta da rede neural, com 20% da amostra destinada a teste. Desta forma, passa-se de um número de regressão  $R=0,91966$  para  $R=0,98414$ .

**Figura 45 - Análise de regressão da rede neural testada para a placa 19 com 1% de imperfeição com alteração dos parâmetros de treinamento**



Fonte: Autoria própria

**Figura 46 - Comparação dos resultados da rede neural testada para a placa 19, com novos parâmetros de treinamento, com resultados obtidos via Abaqus®**



Fonte: Autoria própria

## 5 CONSIDERAÇÕES FINAIS

Mesmo com a configuração ótima, os resultados obtidos pela rede neural treinada com as entradas obtidas via elementos finitos da placa dez e testada com as entradas da placa dezenove, não são satisfatórios. No entanto, observa-se também ser possível treinar uma rede neural com as entradas da placa dez e testando-a com as entradas das placas distintas (quinze e dezessete), obter deformações que se assemelham às encontradas via elementos finitos de maneira aceitável.

Com isso, é possível concluir que para a presente análise, uma rede neural treinada para uma placa específica consegue aproximar resultados satisfatoriamente para placas com orientações angulares similares. Ou seja, a rede neural apresenta bons resultados quando testada para placas que tenham comportamentos estruturais (curva de carga *versus* deslocamento) similares. No entanto, esta mesma rede neural não apresenta resultados satisfatórios quando testada para placas com orientações angulares demasiadamente distintas, que por consequência apresentam curvas de carga *versus* deformação muito diferentes.

Em se considerando o que é apresentado nesta proposta, e que a metodologia do estudo é seguida até a sua última etapa, entende-se que há informações suficientes para fundamentar e validar os resultados encontrados na análise. Ou seja, para o presente estudo de caso, afirma-se que as redes neurais apresentam resultados parcialmente satisfatórios durante a determinação de deformações de uma placa de material compósito laminado submetida ao regime de pós flambagem. Uma alternativa para melhorar os resultados é a realização de treinamento com entradas multiplas. Ou seja, treinar uma rede neural com dados de várias placas. No entanto, a *toolbox* de redes neurais do MATLAB®, *Nftool*, utilizada possibilita treinamentos com apenas uma matriz de entrada e uma matriz de saída.

Ademais, o presente trabalho tem como produto de projeto um tutorial para análise estrutural não-linear utilizando o *software* Abaqus®. O tutorial serve como fonte alternativa de conhecimento para estudantes e professores que desejam fazer a utilização de um *software* CAE, podendo acrescentar estes passos de modelagem em trabalhos acadêmicos.

## REFERÊNCIAS

**BARBERO, Ever J. 2013.** *Finite element analysis of composite materials using Abaqus™*. s.l. : CRC press, 2013.

**DANIEL, Isaac M and ISHAI, Ori. 1994.** *Engineering Mechanics of Composite Materials*. New York : Oxford University Press, 1994, Vol. 3, pp. 12-31.

**Dassault Systèmes®. 2014.** *Abaqus 6.14 Online Documentation*. [Online] 2014. [Cited: junho 05, 2018.] <http://abaqus.software.polimi.it/v6.14/index.html>.

**FALZON, B G e ALIABADI, M. 2008.** *Buckling and Postbuckling Structures: Experimental, Analytical and Numerical Studies*. London : Imperial College Press, 2008, Vol. 1, pp. 20-59.

**FERREIRA, Ana Paula Carvalho da Silva. 2005.** *Identificação do funcional da resposta aeroelástica via redes neurais artificiais*. Universidade de São Paulo. 2005. Dissertação de Mestrado.

**HAYKIN, Simon S. et al. 2009.** *Neural networks and learning machines*. s.l. : Upper Saddle River: Pearson education, 2009.

**KIM, Phil. 2017.** *Matlab deep learning. With Machine Learning, Neural Networks and Artificial Intelligence*. s.l. : Apress, 2017.

**KOVÁCS, Zsolt László. 2002.** *Redes neurais artificiais*. s.l. : Editora Livraria da Física, 2002.

**MONTGOMERY, Douglas C., PECK, Elizabeth A. and VINING, G. Geoffrey. 2012.** *Introduction to linear regression analysis*. Hoboken : John Wiley & Sons, 2012.

**RIKS, Eduard. 1979.** An incremental approach to the solution of snapping and buckling problems. 1979, Vol. 15, 7, pp. 529-551.

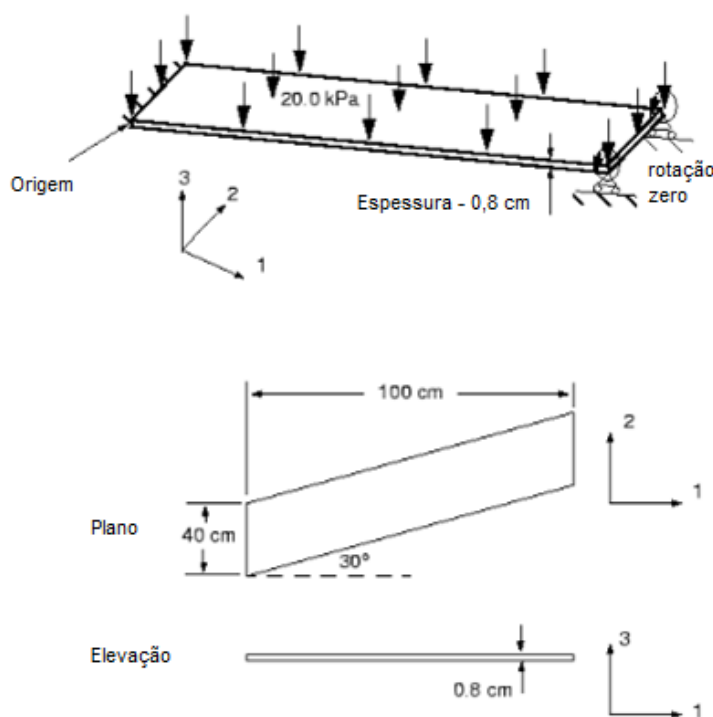
**VASIOS, Nikolaos. 2015.** *Non Linear Analysis of Structures. The Arc Length Method: Formulation, Implementation and Applications*. [Online] 2015. [Citado em: 21 de maio de 2018.] <https://scholar.harvard.edu/files/vasios/files/ArcLength.pdf>.



## APÊNDICE A - ANÁLISE NÃO LINEAR DE UMA PLACA ISOTRÓPICA INCLINADA

A placa inclinada é usada na análise não linear e tem sua geometria, carregamento e condições de contorno apresentadas na Figura 47. Trata-se de uma placa esbelta, fixa em suas duas extremidades, submetida a um carregamento de 20kN distribuído uniformemente em uma de suas faces. A análise envolve definição da geometria e material no programa Abaqus®. Além de condições de contorno, carregamento e malha. A partir disso se obtém e analisa os resultados encontrados. Essas etapas estão descritas nas subseções seguintes.

Figura 47 - Geometria do modelo utilizado

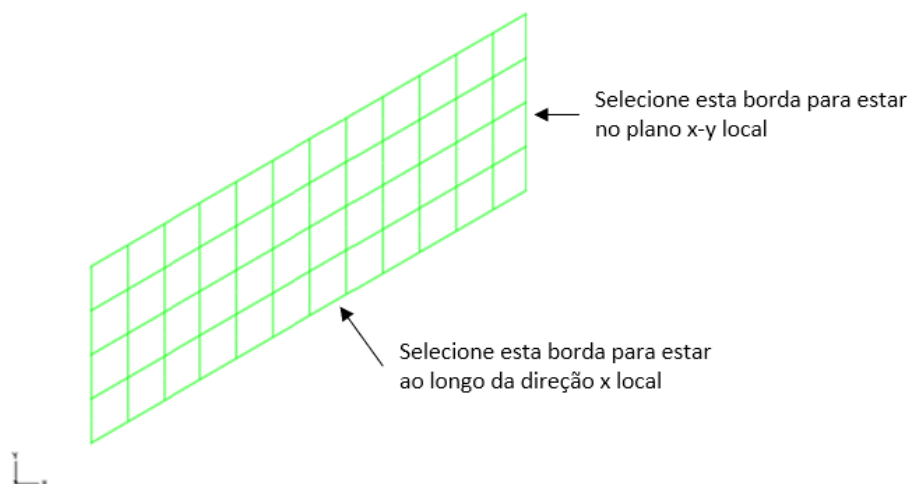


Fonte: Adaptado de (Dassault Systèmes®, 2014)

Primeiramente, para definir a geometria do modelo. Inicia-se o programa Abaqus®/CAE® e cria-se um corpo deformável tridimensional com o recurso de casca planar. Em seguida definem-se as cotas das arestas e o ângulo de inclinação no esboço.

Posteriormente, define-se o material e a orientação local da estrutura. A placa é determinada como sendo uma estrutura isotrópica, elástico linear, com modulo de Young  $E = 30 \text{ GPa}$ , coeficiente de Poisson  $\nu = 0,3$  e com uma espessura de casca de 8 mm. Em seguida, define-se um sistema de coordenadas retangular utilizando a ferramenta: *Create Datum CSYS: 2 lines*

**Figura 48 - Sistema de coordenadas de referência para definir as direções locais do material**

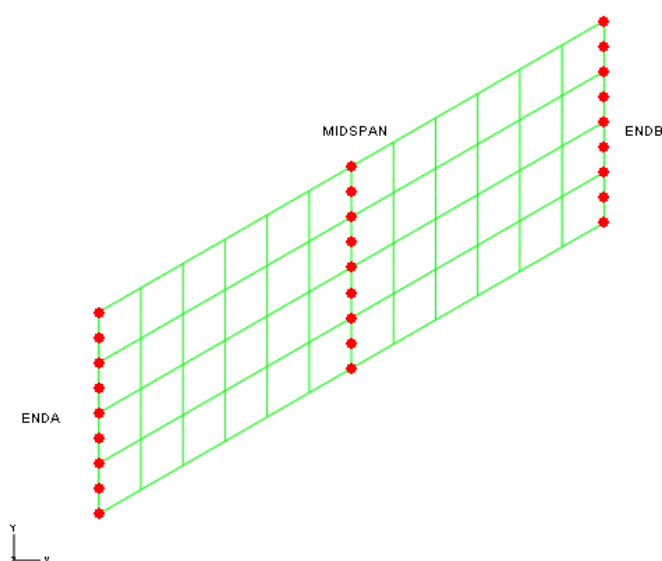


**Fonte: Adaptado de (Dassault Systèmes®, 2014)**

Em *Property*, seleciona-se *Assign* -> *Material Orientation* e define-se a estrutura inteira como região a qual as direções locais do material serão aplicadas. Na janela de exibição seleciona-se o sistema de coordenadas criado anteriormente (selecionando o Eixo 3 como sendo a direção normal da casca).

Para facilitar algumas análises de dados de saída, a estrutura será particionada ao meio, define-se dessa maneira 3 seções (*EndA*, *MidSpan* e *EndB*). Para particionar, utiliza-se a ferramenta *Partition Face: Use Shortest Path Between 2 Points*. Utiliza-se os pontos médios das bordas inclinadas da placa para criar a divisão, como mostrado na Figura 49.

**Figura 49 - Seções criadas para definir os conjuntos de geometrias**



**Fonte: (Dassault Systèmes®, 2014)**

Em *Assembly* cria-se um conjunto de geometria (*Geometry Set*) para o *MidSpan*, para o *EndA* e para o *EndB*. Posteriormente, gera-se um *Step* selecionando as especificações de *single static*, *general step* e aceita-se as configurações padrões para o *Step*.

Para atribuir as condições de contorno e a carga aplicada, é necessário notar que o lado esquerdo da placa está completamente fixo, o lado direito está limitado a se mover em trilhos paralelos ao eixo da placa.

Nesse exemplo, as condições de contorno serão atribuídas em conjuntos (*sets* criados anteriormente) e não em regiões. Para o *EndB*, deve-se fixar todos os graus de liberdade, com exceção do U1. Para o *EndA*, fixa-se todos os graus de liberdade.

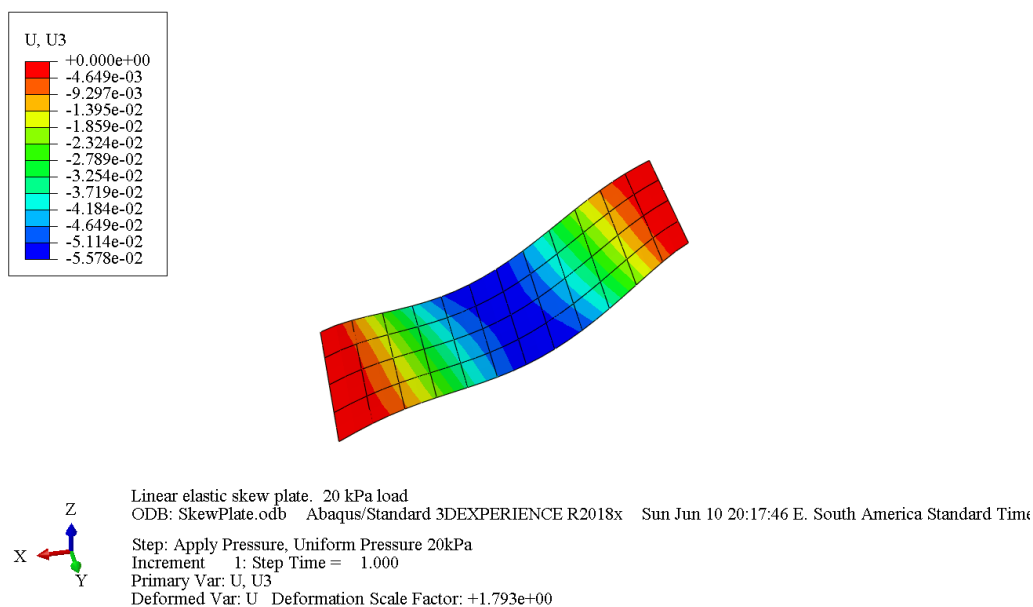
Finalmente, define-se uma carga de pressão uniforme através do topo da casca, selecionando as duas regiões da estrutura. Quando for solicitado que se escolha um dos lados da casca ou face interna, seleciona-se *Brown*, que corresponde ao lado superior da placa. Por fim, especifica-se uma carga de 20 kPa.

Para se definir o tipo de elemento de malha, é preciso determinar se a placa é fina ou espessa e se as deformações são pequenas ou grandes. A placa é fina (8 mm), no entanto não se sabe ainda a magnitude da deformação, mas prevê-se que ela seja pequena. Baseando-se nisso escolhe-se um elemento de casca quadrático (S8R5), devido a seus resultados precisos para elementos de casca finos em simulações de pequenas deformações. Divide-se a malha utilizando um elemento global de tamanho 0,1. Assim sendo, cria-se um *Job* e submete-o a análise e monitorar seus resultados.

Em *Model Tree*, modifica-se o *Set* “*Apply pressure*”, ativando a opção “*NIgeom*” na guia *Basic*, incluindo assim os efeitos de não-linearidade geométrica. Na aba *Incrementation*, define-se como tamanho inicial do incremento 0,1 e mantém-se o número máximo de incrementos em 100, como recomendado pelo manual do usuário.

Na análise linear, o programa resolve as equações de equilíbrio uma vez, calculando os resultados para essa solução, apresentada na Figura 50. Uma análise não-linear pode produzir muito mais resultados, pois os mesmos podem ser solicitados no final de cada incremento convergido.

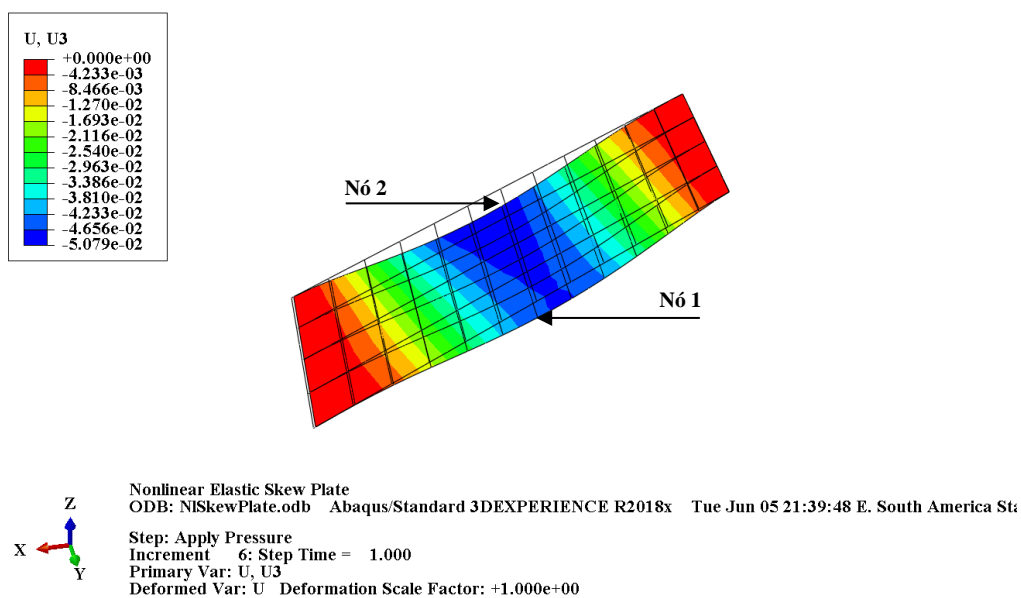
**Figura 50 - Forma deformada e resultado obtido na análise linear**



**Fonte: Autoria própria**

A Figura 51, apresenta as formas deformada e não deformada da placa após a análise utilizando as modificações citadas anteriormente para inclusão dos efeitos de não-linearidade, assim como a magnitude da deformação na direção Z (U3), sendo a escala dada em metros.

**Figura 51 - Forma deformada e não deformada da placa analisada não-linearmente**

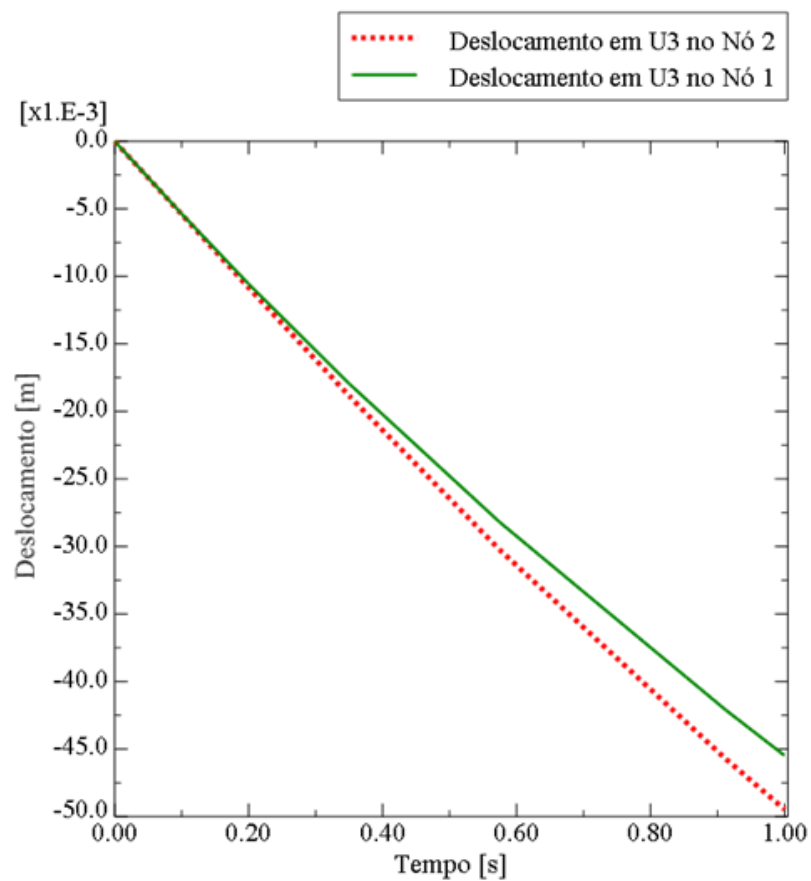


**Fonte: Autoria própria**

A Figura 52 apresenta o gráfico entre o deslocamento na direção Z (U3), em metros, conforme o tempo de análise para os nós 1 e 2, indicados na Figura 51. A natureza não-linear desta simulação pode ser vista nestas curvas, pelo fato de que com o progresso da análise, a placa fica mais rígida. Nesta simulação, o aumento na rigidez da placa com a deformação deve-

se aos efeitos de membrana. Portanto, o deslocamento resultante máximo é menor que o previsto pela análise linear, que não incluiu esse efeito.

**Figura 52 - Deslocamento nas extremidades da região central da placa não-linear**



**Fonte: Autoria própria**