

Implementação da norma brasileira de aço em um programa MATLAB orientado a objetos

Ronald Junior L. Assunção¹, Rafael L. Rangel², Pedro C. Lopes³, Antonyony C. J. Heitor¹, Luiz Fernando Martha¹

¹Department of Civil and Environmental Engineering and Tecgraf Institute of Technical-Scientific Software Development of PUC-Rio
R. Marquês de São Vicente, 225 - Gávea, Rio de Janeiro - 22451-900 - RJ, Brazil
ronaldjunior2@hotmail.com, antonyony@outlook.com, lfm@tecgraf.puc-rio.com

²International Centre for Numerical Methods in Engineering (CIMNE)
C. Gran Capità, S/N 08034 Barcelona, Spain
rrangel@cimne.upc.edu

³Institute of Computing, Fluminense Federal University
Av. Gal. Milton Tavares de Souza s/n - São Domingos, Niterói - 24210-346 - RJ, Brazil
pedrocortez@id.uff.br

Abstract. The complexities in designing structures require that engineers be experts in the use of structural design software. However, many of these programs have complex user interfaces and closed codes. For this reason, it is often difficult for engineering students to use such programs as educational tools. In this regard, the Tecgraf/PUC-Rio Institute is developing an interactive graphics open-source tool on MATLAB for structural analysis of 3D frame structures: the LESM program, which has an intuitive graphical user interface and well-structured code based on the Object-Oriented Programming (OOP) paradigm. This work describes an extension of LESM for the design of steel structures based on the Brazilian standard NBR8800:2008. The code architecture was conceived with modularity, in an abstract OOP fashion, making it possible to extend the program to handle different steel design codes. The definition and organization of the OOP classes and methods that were added to the program are presented, in addition to a demonstration of the usability of the software.

Keywords: steel design, NBR8800, interactive graphics application, MATLAB, object-oriented programming.

1 Introdução

A análise de estruturas metálicas parte da premissa de que os perfis utilizados na edificação devem suportar os esforços solicitantes, atendendo aos padrões normativos da região em que for executada. No Brasil, a norma que rege o uso da maioria das seções metálicas presentes no mercado brasileiro é a NBR8800:2008 (*Projeto de estruturas de aço e estruturas mistas de aço e concreto de edifícios*) [1]. Essa norma define escopos metodológicos de cálculo para cada tipo de esforço solicitante (tração, compressão, flexão simples, flexão composta, que são verificações relativamente triviais. Contudo, caso a edificação analisada possua diversos elementos estruturais ou uma grande variação de seções, o procedimento se torna longo e com muitas variáveis. Por isso, softwares de cálculo estrutural ganham cada vez mais espaço na engenharia e são necessários para que haja um aumento na velocidade de execução e acurácia dos projetos. Observando essa tendência, os autores implementaram uma nova funcionalidade para dimensionamento de perfis metálicos, seguindo as recomendações da NBR8800:2008, no software educacional LESM (*Linear Elements Structure Model*) [2]. Este trabalho tem como principal objetivo descrever a estrutura organizacional das classes, no contexto de Programação Orientada a Objetos (POO), que compõe essa nova funcionalidade. Tais classes foram concebidas de forma que o código disponha de modularidade que permite que outras normas técnicas sejam adicionadas com facilidade. O uso da nova funcionalidade também será brevemente descrito.

2 Referencial teórico

2.1 Normas e dimensionamento

O dimensionamento ou cálculo estrutural é a fase na qual são definidas as dimensões dos elementos da estrutura e suas ligações de maneira a garantir a segurança e o bom desempenho da estrutura [3]. Seguindo esse preceito, países conceberam códigos normativos com o intuito de diminuir desarmonias sobre os procedimentos utilizados e buscar uma padronização das metodologias de cálculo adotadas [4]. Como o escopo desse artigo se apresenta em torno do dimensionamento de perfis metálicos utilizados no Brasil, será utilizada a norma técnica NBR8800 de 2008. Esta norma restringe-se à definição dos princípios gerais que regem o projeto, à temperatura ambiente, das estruturas de aço e das estruturas mistas de aço e concreto [4], não englobando dimensionamento para incêndios e sismos. As metodologias normativas de verificação do dimensionamento dos elementos estruturais são, em geral, procedimentos sequenciais que podem ser programadas em rotinas computacionais.

2.2 Programação Orientada a Objetos

Com o avanço tecnológico ao longo dos anos, cada vez mais a programação é utilizada em múltiplas áreas da engenharia. Dessa forma, linguagens de alto nível são desenvolvidas e atualizadas frequentemente, visto que estas proveem maior afinidade semântica entre a linguagem humana e o código, facilitando a adesão de engenheiros ao desenvolvimento de programas computacionais. Isso permite que os profissionais de engenharia civil, e afins, não dependam de terceiros para o desenvolvimento e customização de rotinas de programação para automatizar os seus cálculos.

Um paradigma de programação muito utilizado e que, certamente, é um facilitador no desenvolvimento de software é a Programação Orientada a Objetos (POO). Esse conceito consiste no desenvolvimento de um programa através da interação entre diversas unidades menores de software, as quais são nomeadas *objeto* [5]. Os objetos são instanciados a partir de classes, que possuem propriedades e métodos. No entanto, após serem instanciados, cada objeto se torna uma unidade individual, com valores independentes entre si, mesmo que sejam originados de uma classe em comum. As principais características da POO, exploradas neste trabalho, são:

Herança: Mecanismo de derivação de classes, onde subclasses são definidas a partir de superclasses. Propriedades e métodos de uma superclasse podem ser passados às suas subclasses, que ainda podem ter suas características específicas. Este conceito permite que o código seja modularizado e reaproveitável.

Abstração: Consiste na generalização da implementação de métodos em uma relação hierárquica. Desta forma, é possível declarar métodos abstratos em uma determinada superclasse, sem os restringir a uma única implementação. É responsabilidade das subclasses herdadas de uma superclasse abstrata a implementação particular dos métodos abstratos.

Polimorfismo: Trata de conceder uma nova implementação aos métodos que são herdados de uma superclasse, sendo possível reutilizá-los com a mesma nomenclatura.

2.3 Organização UML

A UML (*Linguagem de Modelagem Unificada*, em inglês) é uma linguagem visual utilizada para modelar softwares baseados no paradigma de orientação a objeto [6]. Esta define padrões para a elaboração de diversos tipos de diagramas que permitem representar diferentes aspectos e componentes do código, e como eles se relacionam entre si. Os diagramas UML são de grande importância na documentação e publicação de um programa, pois auxiliam o entendimento do mesmo. Exemplos de diagramas UML são o diagrama de Classes, de Sequência, de Objetos, de Pacotes. O diagrama de Classes será utilizado neste artigo para mostrar a estrutura geral do código do LESM com a nova funcionalidade de dimensionamento de perfis metálicos.

3 Metodologia

Para que a implementação dos procedimentos da NBR8800:2008 seja realizada de maneira efetiva no programa LESM, é de suma importância a consideração de boas práticas de desenvolvimento. Assim, além de garantir que os cálculos sejam precisos e condizentes com a norma brasileira, deve-se garantir também que o código seja organizado, de fácil entendimento e, principalmente, sustentável e extensível. Por se tratar de um programa de código aberto e de cunho educacional, a estrutura de dados deve estar preparada para receber novas funcionalidades da comunidade acadêmica, sendo essencial atentar-se à modularidade e ao fluxo de dados.

3.1 Arquitetura de classes

Para o desenvolvimento deste trabalho, foi necessário alterar a estrutura de classes do programa. Um diagrama de Classes, no padrão UML, foi elaborado para ilustrar essas modificações e está mostrado na Fig. 1.

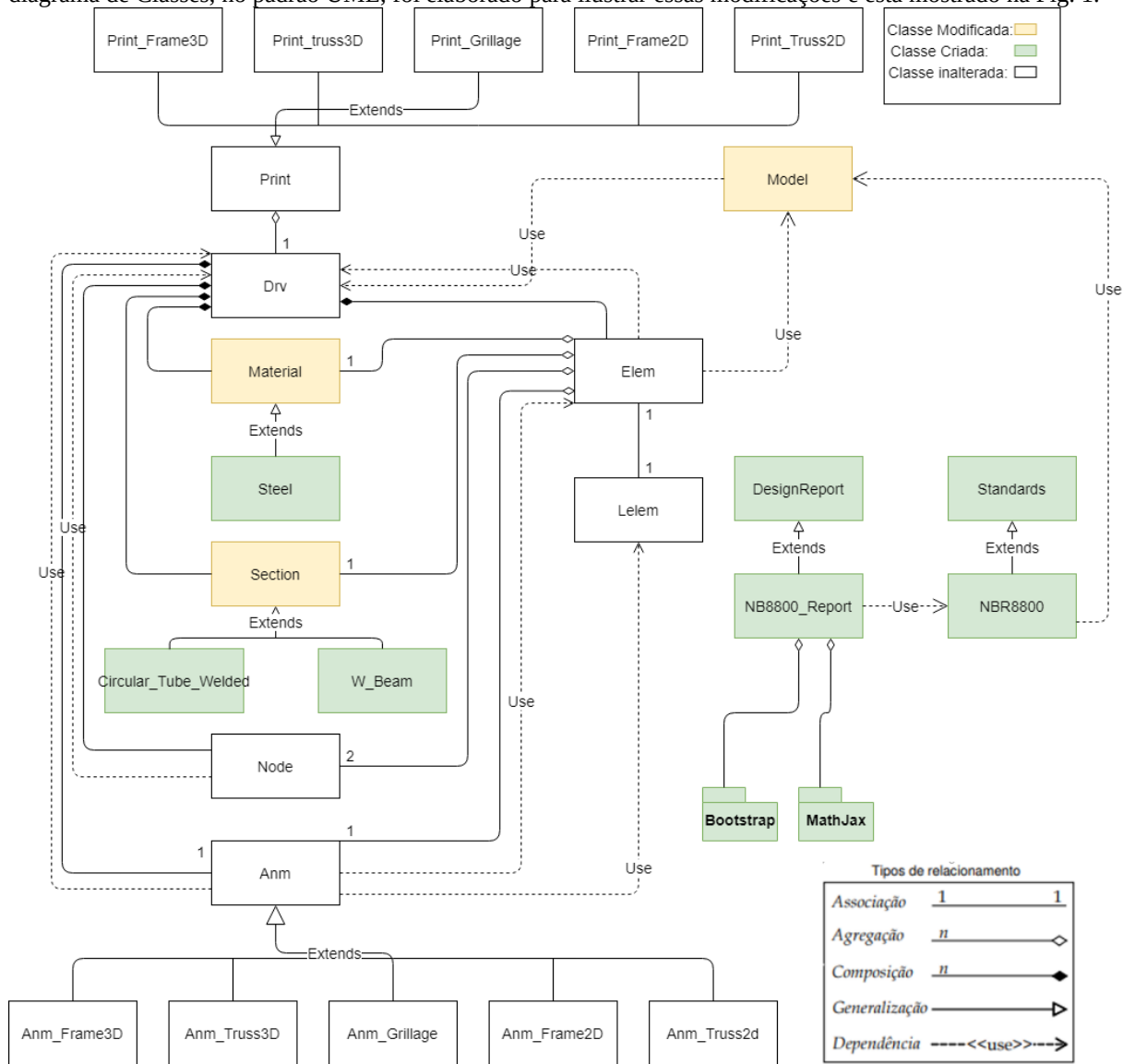


Figura 1. Digrama de classes do programa LESM

A classe **Material** foi abstraída com intuito de possuir apenas propriedades genéricas dos materiais, como densidade, módulo de elasticidade, coeficiente de Poisson, tensão de escoamento. Por herança, a classe **Steel**

recebe todas as características da superclasse e, além disso, possui propriedades próprias como os tipos de aços listados (ASTM A36 e A570, por exemplo) com seus respectivos valores para as propriedades herdadas.

Foi realizada também a abstração da classe **Section** que, por sua vez, possui as propriedades necessárias para cálculos de dimensionamento como área bruta, inércia, coeficiente de torção, entre outras. Esta abstração tem o intuito de criar uma lista dos tipos de perfis mais utilizados comercialmente na construção de edificações metálicas. Dessa forma, foram criadas as subclasses **Circular_Tube** e **W_Beam**, que herdam as propriedades da superclasse **Section**. Estas são classes com listas de famílias de perfis circular tubular e vigas do tipo I, respectivamente.

Após a definição de seções e materiais, o fluxo de dados entre as classes segue o mesmo esquema da versão anterior do programa. Informações do modelo, como os elementos que compõem a estrutura, carregamentos e condições de apoio, assim como os resultados da análise, são salvas na classe **Model**, a qual será utilizada posteriormente na classe **NBR8800**.

A nova classe **NBR8800** é uma subclasse que herda as propriedades da superclasse **Standards**, criada com a finalidade de armazenar informações e implementar métodos comuns a diferentes tipos de normas, de forma que o software fique preparado para receber novos códigos normativos futuramente. A subclasse **NBR8800** é responsável por percorrer cada elemento e verificar se os esforços solicitantes obtidos como resultados da análise são suportados, considerando a metodologia de verificação da norma brasileira de aço. Na sequência, um objeto da subclasse **NBR8800_Report**, herdada da superclasse abstrata **DesignReport**, gera um memorial de cálculo das verificações feitas nos elementos. Este objeto é instanciado por outro, da classe **NBR8800**. Por fim, é gerada uma saída de dados através de um arquivo HTML que registra o memorial de cálculo. Para a visualização desse arquivo, são utilizados dois pacotes web complementares, o **Bootstrap**, para formatação do layout, e o **MathJax** para formatação das equações.

3.2 Rotina de dimensionamento

Rotinas baseadas na norma brasileira de aço foram desenvolvidas para verificar parâmetros de projeto como tração, compressão, flexão simples, flexão composta. Estas estão localizadas na classe **NBR8800**. Como exemplo, a rotina para verificação da compressão em perfis compactos, no qual o parâmetro Q (fator ligado flambagem local) vale 1, é mostrada na Fig. 2. Neste esquema é possível ter uma breve idealização de como todos as rotinas de cálculo foram previamente desenvolvidas antes de serem implementadas.

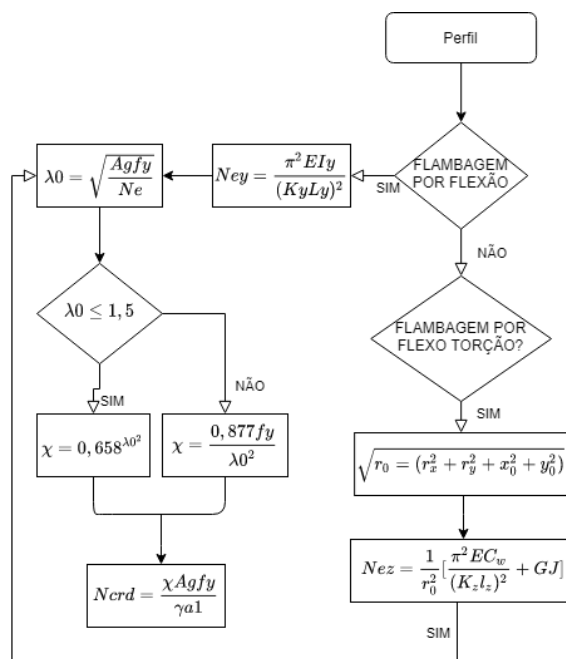


Figura 2. Rotina de verificação à compressão de perfis compactos segundo a NBR8800

3.3 Interface gráfica

Por se tratar de um programa gráfico interativo, a interface do usuário também teve que ser adaptada durante o desenvolvimento deste trabalho para incluir as novas funcionalidades. A Fig. 3 mostra os novos aspectos da interface gráfica do LESM, incluindo o diálogo para a criação de material (1), com a lista de tipos de aços disponíveis, o diálogo para criação de seção transversal (2), com os tipos de perfis disponíveis em catálogos no mercado, e o diálogo para a seleção do tipo de norma a ser utilizada no dimensionamento (3).

4 Resultados

Para exemplificar o uso da nova funcionalidade de dimensionamento de estruturas metálicas, será demonstrada a seguir a verificação da resistência à compressão do perfil W150X13.0 engastada na base e livre na extremidade. Considera-se o aço ASTM-A570 e um esforço solicitante de 500 kN. Como o escopo do trabalho não inclui ligações, não é feita a verificação da área líquida. Além disso, vale ressaltar que a combinação de ações não foi realizada, visto que ainda não foi implementada uma classe responsável por essa ação.

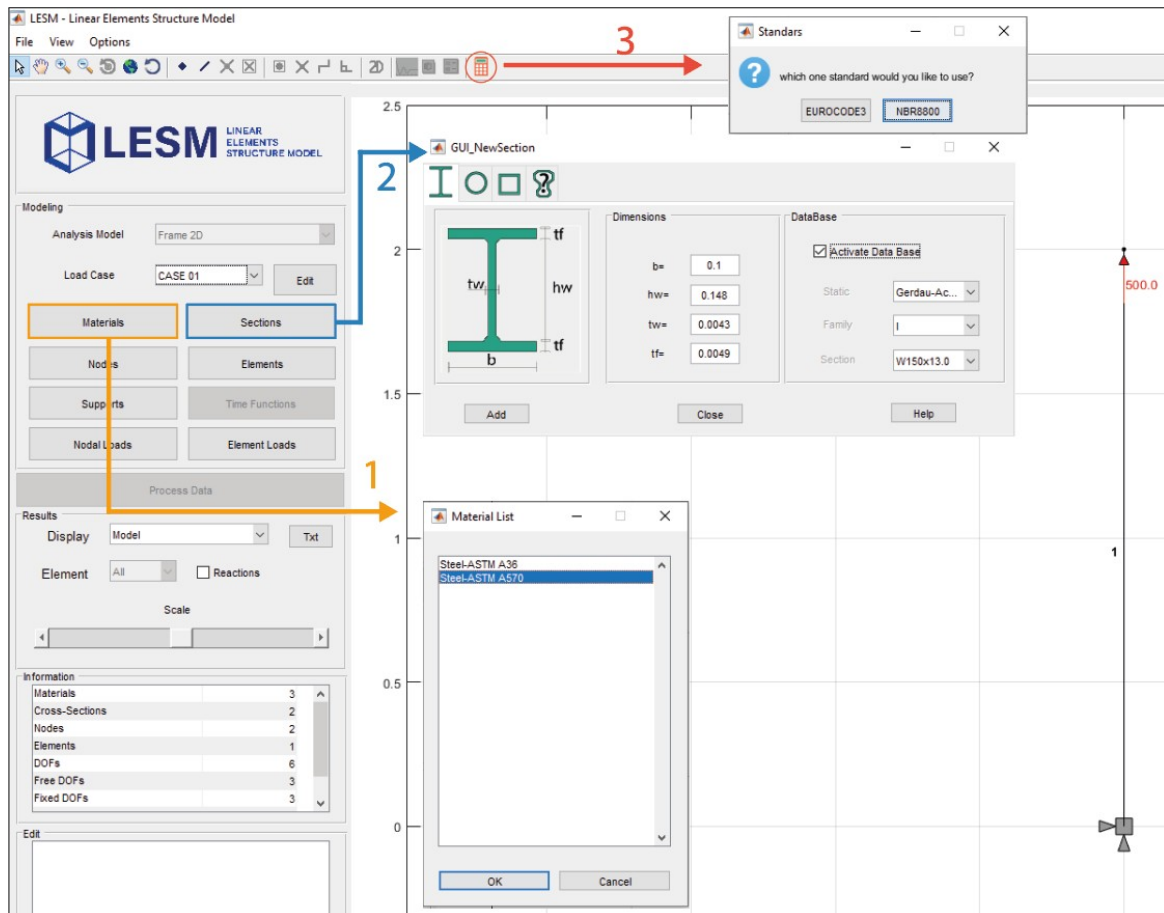


Figura 3. Novos aspecto da interface gráfica do programa LESM

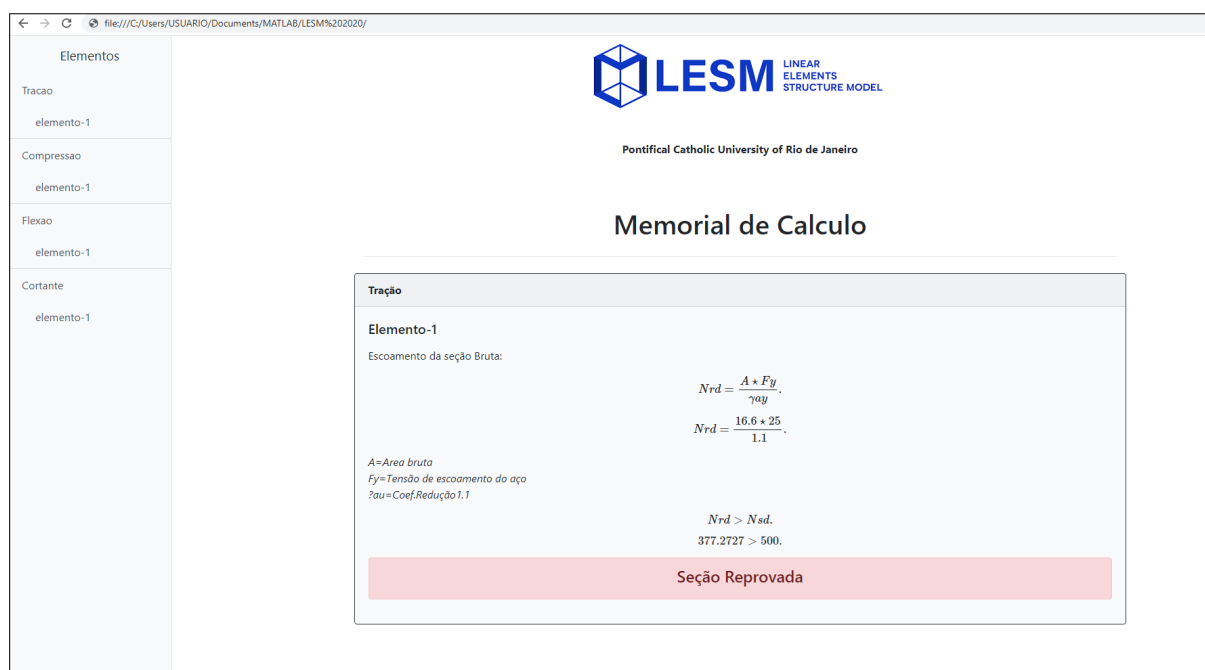


Figura 4. Memorial para cálculo

Observa-se que após o processamento os cálculos são reportados ao usuário associado a cada elemento estrutural, além disso, é exibido se o mesmo resiste ao esforço solicitante através do status aprovado ou reprovado. Neste exemplo o perfil foi reprovado, pois, o valor resistente de projeto é menor que o esforço de tração solicitante (esforço solicitante de projeto).

5 Conclusões

Em suma, a Programação Orientada a Objetos, na qual o software LESM é baseado, é um grande facilitador no processo de acoplamento de novas funcionalidades ao programa, devido ao seu alto nível de abstração. Também fica claro que a POO permite uma melhor estruturação das funcionalidades utilizadas em toda a extensão do projeto. Assim, com a implementação adotando este paradigma, a análise de dimensionamentos de perfis de aço, do ponto de vista metodológico, torna-se simples de ser realizada.

Além disso, espera-se que a implementação destas novas funcionalidades de dimensionamento no software educacional terá um grande impacto positivo aos alunos que o utilizarem, pois o desenvolvimento do mesmo está sendo pautado em possuir uma interface de fácil entendimento e expor detalhes dos cálculos realizados, como pode-se observar na Figura 4.

Tendo em vista os pontos que foram mencionados, vale ressaltar que a versão do código apresentada neste trabalho apresenta uma fase inicial de desenvolvimento do software para dimensionamento de perfis de aço. Existe ainda a necessidade de implementações futuras, como a criação de uma classe de objetos responsáveis pelo gerenciamento das combinações de carga para o cálculo estrutural.

Agradecimentos. Os autores agradecem ao Instituto Tecgraf e ao Departamento de Engenharia Civil e Ambiental da PUC-Rio, pelo incentivo e apoio à pesquisa. Este trabalho foi financiado parcialmente pela Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – CAPES – Código de Financiamento 001.

Authorship statement. The authors hereby confirm that they are the sole liable persons responsible for the authorship of this work, and that all material that has been herein included as part of the present paper is either the property (and authorship) of the authors, or has the permission of the owners to be included here.

References

- [1] ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. NBR 8800: Projeto de estruturas de aço e estruturas mistas de aço e concreto de edifícios. Rio de Janeiro, 2008.
- [2] R. Rangel and L. F. Martha, “LESM – An object oriented MATLAB program for structural analysis of linear element models”. *Computer Applications in Engineering Education*, vol. 27, n. 3, pp. 553–571, 2019.
- [3] W. Pfeil. *Estruturas de Aço: Dimensionamento Prático de Acordo com a NBR 8800:2008*. LTC, 2009.
- [4] L. S. Silva, R. Simões, H. Gervásio, P. Vellasco and L. Lima. *Dimensionamento de Estruturas de Aço: Comparação entre o Eurocdigo 3 e a Norma Brasileira NNBR8800*. Eduerj, 2016.
- [5] T. L. Carvalho. *Orientação a Objetos: Aprenda seus conceitos e suas aplicabilidades de forma efetiva*. Casa do Código, 2016.
- [6] G. Guedes. *UML 2: Uma Abordagem Prática*. Novatec, 2018.