

INSTITUTO FEDERAL DE EDUCAÇÃO CIÊNCIA E TECNOLOGIA DE MINAS GERAIS
CAMPUS SANTA LUZIA
BACHARELADO EM ENGENHARIA CIVIL

Renan Bondezan Cunha

ANÁLISE ISOGEOMÉTRICA PARA MODELOS DE ESTRUTURAS PLANAS

Santa Luzia
2024

RENAN BONDEZAN CUNHA

Trabalho de conclusão de curso apresentado ao curso de Engenharia Civil do Instituto Federal de Minas Gerais – *Campus* Santa Luzia, como requisito parcial para a obtenção do título de Bacharel em Engenharia Civil.

Orientador: Hudson Cleiton Reis Pereira

SANTA LUZIA
2024

Dados Internacionais de Catalogação na Publicação (CIP)

C972a Cunha, Renan Bondezan.

Análise isogeométrica para modelos de estruturas planas. /
Renan Bondezan Cunha. - 2025.
28f. : il.

Trabalho de Conclusão de Curso (Bacharelado em
Engenharia Civil) – Instituto Federal de Educação, Ciência e
Tecnologia de Minas Gerais - Campus Santa Luzia, 2025.

Orientador: Prof. Hudson C. R. Pereira.

1. Modelos estruturais. 2. Engenharia estrutural. 3. Análise numérica em
engenharia. I. Pereira, Hudson C. R. II. Instituto Federal de Educação,
Ciência e Tecnologia de Minas Gerais – Campus Santa Luzia, MG. III.
Título.

CDU: 624.04:517.9

Elaborada pela Biblioteca do IFMG campus Santa Luzia

RENAN BONDEZAN CUNHA

Trabalho de conclusão de curso apresentado ao curso de Engenharia Civil do Instituto Federal de Minas Gerais – *Campus* Santa Luzia, como requisito parcial para a obtenção do título de Bacharel em Engenharia Civil.

Orientador: Hudson Cleiton Reis Pereira

APROVADA EM:

BANCA EXAMINADORA

Prof.º Tiago Simão

Prof.º Rafael Avelino Victoria



MINISTÉRIO DA EDUCAÇÃO
SECRETARIA DE EDUCAÇÃO PROFISSIONAL E TECNOLÓGICA
INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DE MINAS GERAIS
Campus Santa Luzia
Diretoria de Ensino, Pesquisa e Extensão
Docência Eixo Tecnologias- TEC 2 (Tecnologias, Estruturas)
Rua Érico Veríssimo, 317 - Bairro Londrina - CEP 33115-390 - Santa Luzia - MG
3136343910 - www.ifmg.edu.br

ATA DE DEFESA DO TCC

Aos cinco dias do mês de fevereiro do ano de 2024, às 17:45 horas, se reuniu de forma virtual por meio do aplicativo Google Meet a banca composta pelo Prof. Hudson Cleiton Reis Pereira (orientador), Instituto Federal de Minas Gerais – Campus Santa Luzia; Prof. Rafael Avelino Victória (avaliador externo), Tiago Simão Ferreira (avaliador interno), para avaliar o Trabalho de Conclusão do Curso de Bacharelado em Engenharia Civil intitulado **Análise Isogeométrica para Modelos de Estruturas Planas**, apresentado pelo discente **Renan Bondezan Cunha**. Após apresentação e arguição, considera-se o estudante aprovado com nota 60,0. O estudante, terá o prazo de 30 dias para realizar as correções sugeridas pela banca e enviar a versão final ao orientador, para que possa ser anexada ao processo do SEI, sendo a verificação das modificações sugeridas de responsabilidade do orientador. O não envio da versão final corrigida implicará na impossibilidade de solicitação do diploma por parte do discente. Nada mais havendo a tratar a defesa foi encerrada às 19:00 horas e eu, Hudson Cleiton Reis Pereira lavrei a presente ata que, após lida e aprovada, foi assinada por todos os membros da banca e pelo estudante.

Santa Luzia, 08 de fevereiro de 2024.



Documento assinado eletronicamente por **Hudson Cleiton Reis Pereira, Professor**, em 08/02/2024, às 14:29, conforme Decreto nº 10.543, de 13 de novembro de 2020.



Documento assinado eletronicamente por **Renan Bondezan Cunha, Usuário Externo**, em 08/02/2024, às 14:34, conforme Decreto nº 10.543, de 13 de novembro de 2020.



Documento assinado eletronicamente por **Tiago Simao Ferreira, Professor**, em 08/02/2024, às 14:41, conforme Decreto nº 10.543, de 13 de novembro de 2020.



A autenticidade do documento pode ser conferida no site <https://sei.ifmg.edu.br/consultadocs> informando o código verificador **1826439** e o código CRC **FAF15568**.

23716.000205/2024-18

1826439v1

SUMÁRIO

Resumo:.....	5
Abstract:.....	5
1 Introdução	6
2 Objetivos	7
3 Justificativas.....	7
4 Desenvolvimento.....	8
4.1 Modelo matemático vs. Modelo Computacional.....	9
4.2 Método dos Elementos Finitos.....	10
4.3 <i>Computer Aided Design</i>	12
4.4 JAVA.....	14
4.5 Análise Isogeométrica.....	15
5 Metodologia.....	17
5.1 Código.....	19
5.1.1 Convenções	19
5.1.2 Funções Básicas.....	20
5.1.3 Curvas	20
5.2 Viga de Bernoulli-Euler	21
6 Resultados	24
7 Conclusão	26
8 Bibliografia	28

RESUMO:

Esse trabalho tem o intuito de descrever a análise Isogeométrica, a sua criação, além de um pouco da história da computação no ambiente da construção civil e da criação do método, como ele rivaliza com o método computadorizado: método dos elementos finitos.

Palavras-Chave: Análise Isogeométrica, Método dos Elementos Finitos, Estruturas Planas.

ABSTRACT:

This final paper has the point to describe the isogeometric analysis, your history of creation and bring some topics about the history of computation in civil construction. And explore how this new method competes with the most used programming method: finite elements method.

Keywords: Isogeometric analysis, Finite Elements Method, Plane Structures.

1 INTRODUÇÃO

Com o advento do computador, tornou-se muito mais fácil elaborar projetos na construção civil, sobretudo os estruturais que, independentemente da complexidade, eram calculados à mão. Inicialmente, diversas abordagens computacionais foram adotadas; entretanto, apenas uma se destacou: o Método dos Elementos Finitos (MEF), que, gradualmente, consolidou-se como o padrão no cálculo estrutural por meio de softwares especializados

Desenvolvido por volta da década de 1950 o MEF baseia-se na subdivisão da estrutura em elementos menores, formando um modelo discretizado fundamentado em funções de aproximação lagrangeanas. A versatilidade deste método e seu constante aprimoramento ao longo dos anos o tornaram amplamente utilizado em diversas áreas da engenharia, fato comprovado pela adoção de ferramentas renomadas como SAP2000, Nastran, SolidWorks e ANSYS.

Apesar de suas incontestáveis vantagens, o MEF apresenta limitações inerentes à utilização de malhas fixas – onde a discretização é realizada por meio de elementos predefinidos (como triângulos ou quadrados) – o que pode ocasionar imprecisões ao modelar objetos com contornos curvos, exigindo uma divisão mais refinada para garantir a precisão desejada. O fenômeno conhecido como travamento (locking) ilustra bem esse desafio, pois pode comprometer a exatidão dos resultados obtidos

Diante dessas limitações, novos métodos numéricos surgiram para contornar tais desafios, entre eles os Métodos Sem Malha e, principalmente, a Análise Isogeométrica (IGA – Isogeometric Analysis), anteriormente chamada de Método dos Elementos Finitos Isogeométricos. Embora compartilhe semelhanças com o MEF, a IGA propõe uma abordagem inovadora ao integrar de forma natural a modelagem geométrica e a análise estrutural.

Essa abordagem, embora ainda recente – criada em 2005 pelos pesquisadores Hughes T. J. R., J. A. Cottrell e Y. Bazilevs – nasceu a partir da identificação de duas limitações cruciais do MEF: a necessidade de importar desenhos de outros programas e a desconexão entre a representação geométrica e o objeto a ser analisado, problemas decorrentes da re-parametrização das geometrias para adequação ao software de cálculo.

E propuseram, como solução, a integração do desenho assistido por computador, CAD (*Computer Aided Design*), a um método semelhante aos elementos finitos, tornando assim a análise geométrica e estrutural mais próximas entre si, já que assim seria criada a geometria, sua discretização e seu sistema de carregamentos de forma unificada, conseguindo em seus resultados evitar as re-parametrizações geométricas, o que por consequência, aumentou sua eficiência e evitou os já citados problemas.

Em síntese, a Análise Isogeométrica permite a avaliação exata da geometria de problemas mecânicos com maior eficiência, uma vez que o modelo é criado e analisado dentro do mesmo ambiente computacional, garantindo compatibilidade superior e resultados mais consistentes.

Além disso, essa técnica apresenta vantagens significativas em relação ao MEF, como melhor desempenho em otimizações geométricas, mitigação do problema de *locking*, maior eficiência na manipulação de geometrias complexas e irregulares, e uma integração mais direta com as ferramentas CAD e de simulação. Assim, o presente trabalho visa consolidar essas informações e verificar a capacidade da Análise Isogeométrica como um método versátil, promissor e cada vez mais popular na engenharia estrutural.

2 OBJETIVOS

Os objetivos de forma geral deste presente trabalho serão mostrar a evolução computacional para as soluções de cálculos estruturais na construção civil durante a história e quais pontos ainda se pode obter novas soluções e melhoras, estruturar e verificar a elegibilidade da Análise Isogeométrica em relação a outros métodos e sua aproximação em relação ao modelo matemático e ainda trazer material em língua portuguesa sobre a análise Isogeométrica.

Para isso, será elaborado um programa computacional com o objetivo de verificar a facilidade da programação, e testar a aplicabilidade da análise Isogeométrica em relação a um modelo matemático.

3 JUSTIFICATIVAS

Indiscutivelmente, para a engenharia civil, o método dos elementos finitos é uma das metodologias de cálculo estrutural computacional mais importantes, sendo

utilizado pela maioria de softwares. Mas, como toda metodologia, ela apresenta pontos de negativos seja por suas falhas ou a sua não otimização para certos casos. Os principais pontos são: seu desempenho não tão eficiente em situações de otimizações geométricas (*shape optimization*), os travamentos (*Locking*), em problemas que necessitam de um ajuste fino de continuidade e o principal motivo pela criação da análise que será objeto de estudo desse trabalho, a separação entre software de modelagem e o software de análise estrutural.

Visto isso, a análise Isogeométrica é um método que consegue ser extremamente parecido com o MEF, mas, que otimiza nos pontos explorados anteriormente. E, com a evolução de projetos com a utilização de BIM, torna-se extremamente importante a maior compatibilidade entre a modelagem de uma estrutura e seu dimensionamento estrutural, e, também, causa a necessidade de se repensar os métodos já utilizados no mercado, para que se consiga pensar em uma melhoria contínua na modelagem estrutural.

4 DESENVOLVIMENTO

O método de análise Isogeométrica foi desenvolvido por Hughes T. J. R, J. A. Cottrell e Y. Bazilevs em seu artigo:” *Isogeometric analysis: CAD, Finite elements, NURBS, exact geometry and mesh refinement*” de 2005. Em seu texto, a partir de um breve histórico do método dos elementos finitos, criado na primeira metade do século XX, o autor nota que o método é desconexo geometricamente dos elementos calculados, além da necessidade de utilizar outra ferramenta para a modelagem do objeto.

Então, de forma resumida, para solucionar, adotou-se como uma alternativa uma implementação do CAD (*Computer Aided Design*) a um método similar ao Elementos Finitos, assim conseguindo discretizar o objeto independentemente de sua geometria, mesmo que de forma turbulenta, como o próprio Hughes cita. Como objetos de testes para o estudo foram utilizados objetos tridimensionais, o que por consequência aumenta sua complexidade: estruturas cilíndricas lineares, cascos e fluídos. E, como resultado, esperava-se que junto a uma aproximação geométrica mais precisa, a análise de esforços fosse mais aprimorada do que o outro método.

No entanto, com os resultados, os autores perceberam que ao unir CAD a uma metodologia de cálculo computacional, houve mais vantagens computacionais do que

apenas a maior precisão e eficiência com método. Em um dos resultados, utilizando fluídos, por sua forma não tão bem definida, conseguiu uma análise precisa com menor grau de refinamento e maior velocidade.

Ao trazer para engenharia civil, por exemplo, utilizando o método em estruturas em arco, típica forma que tende a ter um custo computacional extremamente alto quando utilizado o Método dos Elementos Finitos, por necessitar de alto grau de refinamento; com a Análise Isogeométrica, o custo computacional, o refinamento e o tempo para se analisar uma estrutura arqueada é extremamente menor.

Dado isso, antes de entrar de fato no método e compará-lo com o resultado do modelo matemático, é importante salientar o porquê da existência desses métodos computacionais visto que já há a existência de um modelo matemático preciso para esses mesmos problemas.

4.1 Modelo matemático vs. Modelo Computacional

Para todo problema da construção civil, tem-se uma solução dada pelos modelos matemáticos, por mais complexos que sejam, e estes resultados se darão de forma direta em seu cálculo, sem a necessidade de aproximações ou repetições de um mesmo processo, o que nos gera um processo único para cada problema. Já os modelos computacionais, seja a análise Isogeométrica ou o método dos elementos finitos, como se perceberá, serão executados por aproximações para o resultado real do problema, sendo dado por repetições e refinamentos de um mesmo processo, tornando o método de solução genérico.

As repetições de um mesmo processo, apenas o refinando, é menos custoso e mais veloz para um computador rodar, do que comparado ao modelo matemático, que traz diversas contas e processos diferentes para cada elemento. Em resumo, para se programar um modelo matemático preciso seria necessário criar um programa que fizesse apenas um tipo de elemento, podendo ter problemas de compatibilidade, mesmo com uma diferença mínima, enquanto com os métodos computacionais, além de mais veloz, eles são mais generalistas, podendo calcular, com um mesmo método, vários tipos de elementos.

A partir disso, serão apresentados importantes conceitos, suas histórias de desenvolvimento, e como funcionam, para que se entenda o problema a ser analisado e seus resultados.

4.2 Método dos Elementos Finitos

O MEF, método dos elementos finitos, ou em inglês, FEM, *Finite Element Method*, foi criado em 1909 por Walter Ritz, com o objetivo de calcular uma solução de forma aproximada para problemas de sólidos deformáveis.

Com o decorrer dos anos, outros engenheiros desenvolveram ainda mais o método como, por volta da década de 40, Richard Courant, fez incremento de funções lineares de regiões triangulares, conseguindo aumentar o leque de possibilidades do método, sendo possível ser utilizado agora para problemas de torção. Com essas alterações e outras, Ray William Clough Jr (NAE, 2016), na década de 1960, introduziu de forma definitiva o método, além de mostrar outras aplicações que não tinham sido apresentadas ainda, como análise de estruturas aeroespaciais, fluidos, termodinâmica e eletromagnetismo.

Mas, apenas nos anos de 1950, teve início de fato a utilização do método em estruturas relacionadas a construção civil, principalmente com o desenvolvimento e as discussões relacionadas ao método das forças e o método dos deslocamentos. E, em 1959, Turner cria o Método da Rigidez Direta, que é baseada no segundo método.

Visto isso, o MEF (AZEVEDO,2003) consiste em discretizar domínios, isto é, seccionar objetos em formas geométricas mais simples, geralmente utilizando quadriláteros e triângulos, podendo ser, também, a combinação de ambas as formas. Quanto maior a malha criada a partir dessas formas, mais precisa será o resultado, contudo haverá maior gasto computacional.

A partir disso, Azevedo (2003) descreve, a discretização funciona por três passos: criação da malha, definição das funções de forma e o mapeamento dos elementos de referências da malha. Considerando um elemento genérico unidimensional, P, ser discretizado tem-se que,

$$P: \begin{cases} u''(x) = f(x); & em (0, 1) \\ u(0) = u(1) = 0 \end{cases} \quad (1)$$

Sendo,

O próximo passo será converter P em sua equivalente forma fraca e para isso, integramos por partes utilizando as identidades de Green, portanto tem-se:

$$\int_0^1 f(x)v(x)dx = \int_0^1 u''(x)v(x) dx = u'(x)v(x)|_0^1 - \int_0^1 u'(x)v'(x)dx = - \int_0^1 u'(x)v'(x)dx \equiv \Phi(u, v) \quad (2)$$

Portanto, para discretizar o elemento, tem-se que tomando o intervalo $(0,1)$, para n valores de x , toma-se que $0 = x_0 < x_1 < \dots < x_n < x_{n+1} = 1$, consegue-se definir que V por:

$$V = \{v: [0,1] \rightarrow \mathbb{R}: \text{sendo } v \text{ contínuo, } v|_{x_k, x_{k+1}} \text{ é linear para } k = 0, \dots, n \text{ e } v(0) = v(1) = 0,$$

Sendo V um subespaço definido por H_0^1 . Com isso feito, tem-se a necessidade de escolher a função de base, sendo que para problemas de uma dimensão, para cada ponto de controle x_k , escolhe-se a parte da função v_k em V , sendo então,

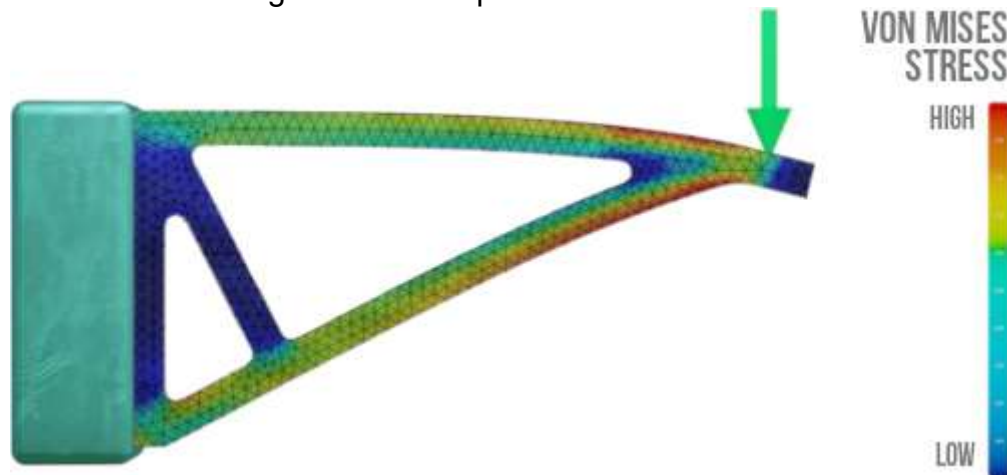
$$v_k(x) = \begin{cases} \frac{x-x_{k-1}}{x_k-x_{k-1}}, & \text{se } x \in [x_{k-1}, x_k], \\ \frac{x_{k+1}-x}{x_{k+1}-x_k}, & \text{se } x \in [x_k, x_{k+1}], \\ 0, & \text{para o restante} \end{cases} \quad (3)$$

Tendo isso, escreve-se as matrizes de forma para o problema, sendo escrito por,

$$u(x) = \sum_{k=1}^n u_k v_k(x) \text{ e } f(x) = \sum_{k=1}^n f_k v_k(x) \quad (4)$$

Com isso, tendo que compararemos objetos bidimensionais, a dedução funciona de forma análoga para a nova direção que aparecer no problema, adicionando assim logo na integral de forma fraca.

Figura 1 – Exemplo de malha no MEF



Fonte: The Efficient Engineer, 2023.

A figura 1, mostra como uma estrutura é subdividida no MEF. Como pode-se perceber, é uma divisão feita por objetos geométricos iguais, o que como já citado anteriormente, causa as imprecisões e a necessidade de aumentar o refinamento da

malha em objetos curvos, para assim conseguir se adequar de forma mais precisa ao objeto.

Além disso, Azevedo (2003), cita que a utilização das formas geométricas triangulares cria imprecisões em relação a distribuição das cargas, criando pontos de pressões perto dos vértices dos triângulos, fazendo com que haja distribuições errôneas.

4.3 Computer Aided Design

O CAD, *Computer Aided Design*, em português, desenho assistido por computador, é a forma de utilizar, através de programação, o computador para desenhar. Sua história, de acordo com a BELLUOMINI (2017), começa antes mesmo da invenção do computador, com Alphonse Louis Poitevin (1819-1882), que em 1861 descobriu um processo que envolvia a sensibilidade a luz da gelatina bicromada, uma substância presente em gomas de mascar, que utilizou para criar os processos de Fotolitografia e fototipia.

Estes processos seriam importantes na construção civil, por ser extremamente precisas em reproduzir desenhos. Consistia em criar cópias em negativo, as famosas “*Blueprints*”, que transformava as linhas pretas em brancas, e os espaços vazios em um tom azulado.

A partir de 1960, cerca de 100 anos após o desenvolvimento da tecnologia de cópias precisas de desenhos arquitetônicos, a BELLUOMINI (2017) conta, que atualmente, com o advento e popularização dos computadores, Patrick J. Harrantly, em 1961, começa o desenvolvimento, na General Motors, do chamado DAC, *Design Automated By Computer*, e que nos anos seguintes seria aperfeiçoado e chamado de CAD.

O processo para se gerar qualquer forma geométrica, de forma mais simples computacionalmente, é utilizando funções matemáticas parametrizadas, geralmente polinômios de interpolação que derivam do polinômio lagrangiano e das Curvas de Bèzier, chamadas Spline. Existem vários tipos de Spline, as principais são: *B-spline*, *T-spline* e *NURBS*. A partir disso, com a escolha de pontos de controle que o usuário toma e forma que se necessita criar, a forma geométrica é criada.

Tomaremos como foco a *B-spline* e *Nurbs*, por serem as mais conhecidas e mais utilizadas na programação, além de terem um excelente poder de convergência

assim como HUGHES (2005) utilizou em seu trabalho. A NURBS, *Non-Uniform Rational Basis Spline* (*Basis Spline Não-Uniforme Racional*), como o próprio nome diz, é derivada da *Basis Spline* (*B-Spline*).

A *B-Spline* consiste em um modelo matemático que gera curvas a partir de três conceitos: grau, suavidade e domínio. Por definição (GAN,2018), uma *Spline* de ordem “ n ” são funções polinomiais por partes de grau $n-1$ com variável x , sendo x a coordenada em que as partes do polinômio coincidem, que são chamados de nós e são dados por $t_0, t_1, t_2, \dots, t_n$.

Portanto, para uma sequência de nós, temos uma única *Spline* dada por,

$$B_{i,n}(x) = \begin{cases} \mathbf{0}, & \text{se } x < t_i \text{ ou } x \geq t_{i+n} \\ \text{não nulo}, & \text{se de outra forma} \end{cases} \quad (5)$$

Com isso, adicionando a restrição de $\sum_i B_{i,n}(x) = 1$ para todo x no intervalo do primeiro e último nó, percebe-se que $B_{i,n}$ terá uma taxa de crescimento fixa, tornando-a assim a chamada *B-Spline*.

Outra forma de construí-la é a partir do algoritmo de Boor, que dada uma sequência de nó t_0, t_1, t_2 , uma B-spline de grau um é dada por:

$$B_{i,1}(x) = \begin{cases} \mathbf{1}, & \text{se } t_i < x < t_{i+1}, \\ \mathbf{0}, & \text{se de outra forma} \end{cases} \quad (6)$$

E assim, para B-splines de graus maiores são definidas por equações de repetição dadas por:

$$B_{i,k+1}(x) = w_{i,k}(x)B_{i,k}(x) + [1 - w_{i+1,k}(x)]B_{i+1,k}(x), \text{ sendo}$$

$$w_{i,k}(x) = \begin{cases} \frac{x+t_i}{t_{i+k}-t_i}, & t_{i+k} \neq t_i \\ \mathbf{0}, & \text{se de outra forma} \end{cases} \quad (7)$$

E, para o NURBS, segundo GAN (2018), sua diferença é dada pela sua não uniformidade já explicitada em seu nome, que é dada pela utilização um sistema de coordenadas homogêneas (x, y, u) , ao invés de um sistema cartesiano. Essas coordenadas consistem na representação do plano cartesiano (X, Y) de forma parametrizada da seguinte maneira: $X = x/u$ e $Y = y/u$. Isso traz uma otimização computacional ao processo por causa de seu tratamento algébrico nos pontos “no infinito”.

Além disso, tem-se também um outro importante conceito introduzido no NURBS, o Knot Vector, que consiste numa sequência de parâmetros que definem quando e como os pontos de controle afetam a curva, sendo o número de Knot Vector

igual ao número de pontos de controle mais o grau da curva menos um. Portanto a curva é dada por:

$$C(u) = \frac{\sum_{i=1}^k N_{i,n}(u)w_i P_i}{\sum_{i=1}^k N_{i,n}(u)w_i} \quad (8)$$

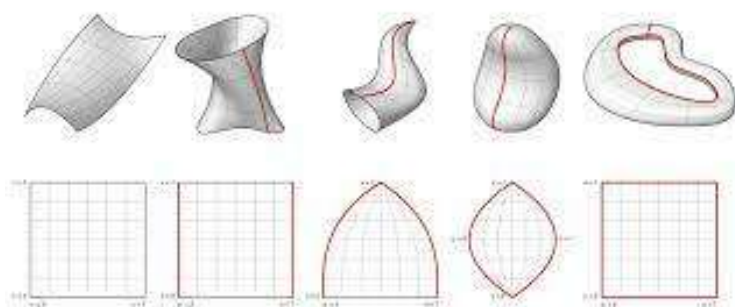
Sendo,

u = uma variável independente; k = número de pontos de controle;

N = B-spline; n = grau do polinômio;

P = ponto de controle w = peso/intensidade;

Figura 2 – Exemplo de NURBS em objetos tridimensionais



Fonte: RAAMAC (2023)

4.4 JAVA

A linguagem que será adotada para o programa de análise Isogeométrica será JAVA, que começou a ser criada em 1991, de acordo BESSA (2023), pela Sun MicroSystems, empresa que posteriormente foi comprada pela Oracle, sua atual desenvolvedora. A Sun começou a sua criação pois seus desenvolvedores não conseguiam fazer todas as programações desejadas no C++, a linguagem mais comum da época. Inicialmente foi lançada com o nome de Oak, em 1993 e, por problemas de *Copyright*, em 1994, teve seu nome alterado para JAVA.

Ela foi projetada com várias características inovadoras que a distinguem de outras linguagens de programação da época. Uma das características mais marcantes do Java foi o conceito de "*Write Once, Run Anywhere*" (do inglês: "Escreva uma vez, execute em qualquer lugar"), que significava que os programas Java poderiam ser compilados em *bytecode* e executados em qualquer dispositivo que tivesse uma

máquina virtual Java (JVM), independentemente da arquitetura do sistema. A portabilidade, segurança e simplicidade da linguagem Java a tornaram extremamente popular entre os desenvolvedores. Ela foi amplamente adotada em uma variedade de aplicativos, desde aplicativos de desktop até sistemas distribuídos e aplicações web.

Ao longo dos anos, a plataforma Java cresceu e se expandiu para incluir uma ampla gama de tecnologias e frameworks, como a *Enterprise Edition* para desenvolvimento de aplicativos corporativos, a *Standard Edition* para aplicativos de desktop e a *Micro Edition* (Java ME) para aplicativos móveis e embarcados.

As principais vantagens da linguagem são suas ferramentas gratuitas, grandes repositórios em sites como o “GitHub”, por ser uma linguagem popular, esteve em 2022 como a 6ª linguagem mais utilizada por programadores, e sua forma de programação é feita orientada a objetos, isto é, há a separação da parte de abstração, do encapsulamento de código e herança, o que facilita a programação e para iniciantes confere uma curva de aprendizagem mais suave. Além de outras vantagens mais técnicas como a execução concorrente de rotinas, criando assim robustez no programa.

4.5 Análise Isogeométrica

Por fim, e mais importante, a análise Isogeométrica tendo sua história já introduzida e conceitualmente iniciada, se terá como foco, portanto, sua definição matemática de forma detalhada mostrada a seguir.

Da mesma forma que o método dos elementos finitos, a análise também trabalha em malhas, que aqui são chamadas de sub-regiões, que diferentemente, trabalha tanto a forma física, quanto sua forma geométrica, e que é formada a partir de pontos de controle que com sua interpolação, forma as sub-regiões.

Essas interpolação, de acordo com GAN (2019), são dadas pelo *Knot Vector*, que é um grupo de coordenadas unidimensionais não decrescentes, indicado por $\Xi = \{\xi_1, \xi_2, \xi_3, \dots, \xi_{n+p+1}\}$, sendo que $\xi_i \in \mathbb{R}$, sendo ξ_i o i -ésimo nó, n o número de funções de controle e p é a ordem da função de base. O *Knot Vector* divide a estrutura em elementos, sendo que eles são uniformes se são equidistantes e não uniformes quando não são.

As funções de base são definidas por

- Para $p = 0$:

$$N_{i,0}(\xi) = \begin{cases} 1 & \text{se } \xi_i \leq \xi \leq \xi_{i+1} \\ 0, & \text{se de outra maneira} \end{cases} \quad (9)$$

- Para $p=1,2, 3$,

$$N_{i,0}(\xi) = \frac{\xi - \xi_i}{\xi_{i+p} - \xi_i} N_{i,p-1}(\xi) + \frac{\xi_{i+p+1} - \xi}{\xi_{i+p+1} - \xi_{i+1}} N_{i+1,p-1}(\xi) \quad (10)$$

Sendo que elas respeitam a relação de:

$$\sum_{i=1}^n N_{i,p}(\xi) = 1 \quad (11)$$

E as derivações das funções de base é dada por:

$$\frac{d^k}{d^k \xi} N_{i,p}(\xi) = \frac{p!}{(p-k)!} \sum_{j=0}^k \alpha_{k,j} N_{i+j,p-k}(\xi) \quad (12)$$

Sendo

$$\alpha_{0,0} = 1$$

$$\alpha_{k,0} = \frac{\alpha_{k-1,0}}{\xi_{i+p-k+1} - \xi_i}$$

$$\alpha_{k,j} = \frac{\alpha_{k-a,j} - \alpha_{k-1,j-1}}{\xi_{i+p+j-k+1} - \xi_{i+j}} \text{ para } j = 1, \dots, k-1$$

$$\alpha_{k,k} = \frac{\alpha_{k-1,k-1}}{\xi_{i+p+1} - \xi_{i+k}}$$

Com isso definido, a geometria é dada, utilizando NURBS, que tendo n funções de base $N_{i,p}$, sendo $i=1, 2, \dots, n$, temos que:

$$R_i^p(s) = \frac{N_{i,p}(s)\omega_i}{W(s)} \quad (13)$$

Com isso, portanto, utilizando a mesmo polinômio por partes que a B-spline, tem-se

$$C(s) = \sum_{i=1}^n R_i^p(s)B_i \quad (14)$$

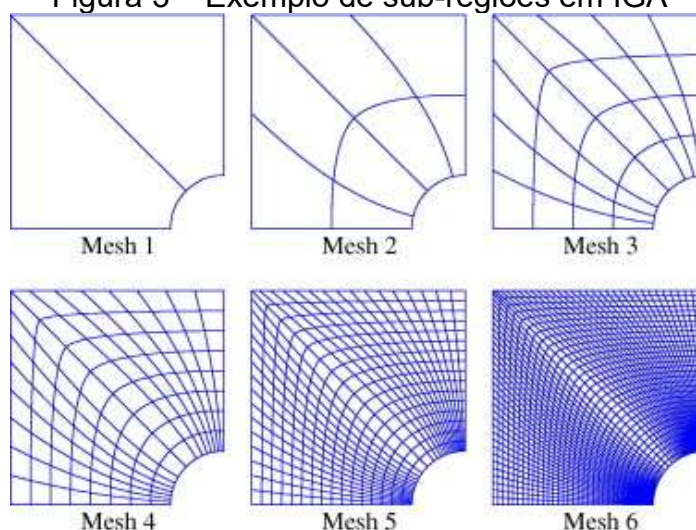
É importante notar, segundo GAN (2019), que para a montagem dos *Knot Vectors* precisa de respeitar algumas regras para determinar uma curva, já que ele determina o intervalo de início e de fim dos parâmetros da curva representada, portanto os seguintes fundamentos são:

- O número de pontos de controle deve ser igual ou maior que a ordem da curva;
- A ordem da curva deve ser maior ou igual a dois
- A quantidade de *knots* no *Knot Vector* é igual ao número de pontos de controle mais a ordem da curva, sendo definido por $n+p+1$. Sendo que os primeiros elementos $p+1$ serão iguais a 0 e os últimos $p+1$ serão

iguais a 1 ou a múltiplos deste. Já os valores intermediários serão definidos de forma a ser equidistantes;

- Os valores no *Knot Vector* não devem ser decrescentes, portanto, um exemplo de disposição válida é $[0,0,0,0.25,1,2,2,2]$ e de uma não válida seria $[0,0,0,0.5,0.25,2,2,2]$
- $\xi_{\min} \leq \xi < \xi_{\max}$, sendo $\xi_{\min} = \xi_{\text{ordem}-1}$ e $\xi_{\max} = \xi_{\text{NC}}$, sendo NC o número de pontos de controle.

Figura 3 – Exemplo de sub-regiões em IGA



Fonte: HUGHES, 2005.

Na figura três, tem-se um exemplo, dado pelo próprio HUGHES (2005), de como se comporta a sub-região que a Análise Isogeométrica cria no objeto, como se percebe, diferentemente do método dos elementos finitos, as sub-regiões criadas se moldam de acordo com o objeto e não o subdivide em objetos iguais, sendo isso de extrema importância para o método, principalmente em análises de objetos tridimensionais curvos, já que sua aproximação geométrica se torna mais precisa.

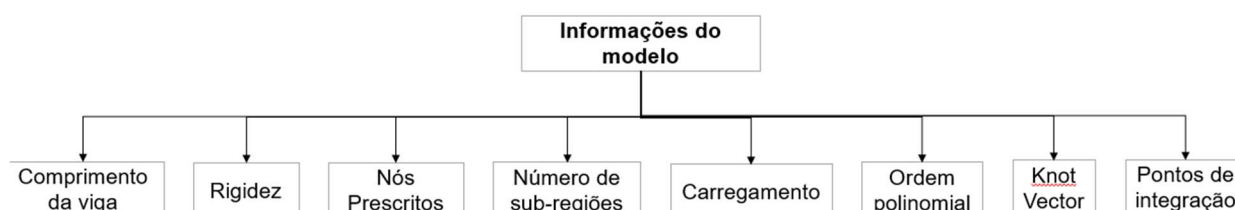
5 METODOLOGIA

Como já comentado anteriormente, para o desenvolvimento deste trabalho será utilizado um código computacional escrito em linguagem Java. Este já estava previamente desenvolvido pelo orientador do trabalho, sendo originalmente desenvolvida em *B-Spline*, optou-se por alterar para NURBS, para melhor otimização computacional e compatibilidade com outros programas *CAD*, já que é o método mais utilizado e é a forma em que foi originalmente pensado.

Vale ressaltar que, o programa desenvolvido é facilmente reproduzível em outra linguagem de programação, já que não há a necessidade de bibliotecas específicas. A opção por Java é dada por sua orientação a objetos e sua extensão de algoritmos, tornando este trabalho menos oneroso além de conhecimentos prévios.

A figura quatro mostra com quais informações o modelo será alimentado:

Figura 4 – Informações que alimentam o modelo



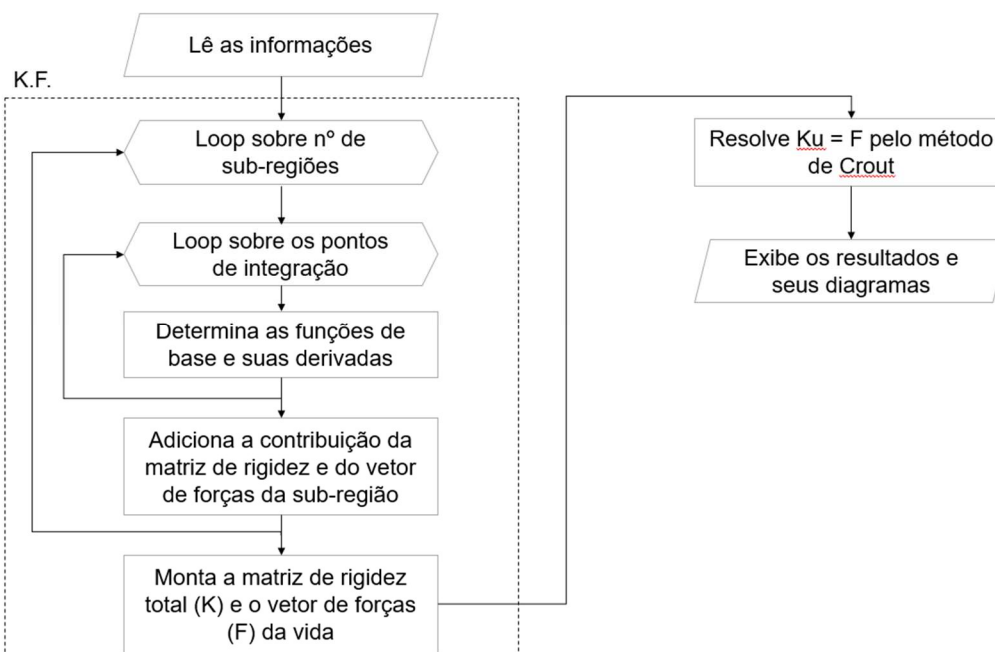
Fonte: Elaborado pelo autor, 2023

Nota-se dois pontos importantes, diferentemente do método dos elementos finitos, não há malha e sim sub-regiões para a divisão do elemento e que a ordem polinomial da função de base é escolhida como dado de entrada, facilitando a precisão do usuário para a solução do problema.

E a seguir, na figura cinco, tem-se o fluxograma das rotinas implementadas. Importante notar que a matriz de rigidez K e a construção do vetor de forças equivalentes F , são duas tarefas independentes que serão executadas em paralelo o que causará um consumo maior de processamento no momento, mas que diminuirá o tempo de execução do programa.

Observa-se também que, uma vez que a matriz de rigidez é positiva definida, será garantida a sempre existência de uma inversa e fazendo assim com que sempre haja a solução do sistema de equações $Kw=F$ obtida usando o método de Crout.

Figura 5 – Fluxograma das rotinas



Fonte: Elaborado pelo autor, 2023

5.1 Código

Com a estrutura definida do programa e utilizando os conceitos de NURBS apresentados anteriormente, junto do auxílio do código já executado pelo orientador Hudson, o pseudocódigo para a mudança de b-spline para NURBS, será definida em três etapas: convenções, as funções básicas e curvas. Como o problema a ser resolvido é dado em um plano bidimensional, não haverá necessidade de funções para superfícies, com isso:

5.1.1 Convenções

Grau: inteiro p (curvas) e q (superfícies).

Vetores de nós: $U = [u_0, \dots, u_n]$ e, para superfícies, $V = [v_0, \dots, v_n]$.

Pontos de controle: $P[0, n]$ (curvas) ou $Q[i][j]$ (superfícies), dimensionais (2D).

Pesos: $w[0, n]$ ou $w[i][j]$, todos positivos.

Parâmetros de avaliação:

Para curvas: $u \in [U[p], U[n-p]]$;

para superfícies $v \in [V[q], V[n-q]]$

A curva NURBS é dada por

$$C(u) = \frac{1}{\sum_{i=0}^n N_{i,p}(u) w_i P_i} \times \frac{1}{\sum_{i=0}^n N_{i,p}(u) w_i}$$

5.1.2 Funções Básicas

```
# Encontra o span (índice i tal que  $u \in [U[i], U[i+1])$ )
function FindSpan(n, p, u, U):
  if u == U[n+1]:
    return n
  low = p
  high = n + 1
  mid = (low + high) // 2
  while (u < U[mid]) or (u >= U[mid + 1]):
    if u < U[mid]:
      high = mid
    else:
      low = mid
  mid = (low + high) // 2
  return mid
```

```
# Funções base B-spline  $N_{\{i-p..i, p\}}(u)$  (Cox-de Boor)
function BasisFuns(i, u, p, U):
  N = array[0..p] filled with 0
  left = array[1..p]
  right = array[1..p]
  N[0] = 1.0
  for j in 1..p:
    left[j] = u - U[i + 1 - j]
    right[j] = U[i + j] - u
    saved = 0.0
    for r in 0..(j-1):
      denom = right[r + 1] + left[j - r]
      if denom == 0:
        temp = 0.0
      else:
        temp = N[r] / denom
      N[r] = saved + right[r + 1] * temp
      saved = left[j - r] * temp
    N[j] = saved
  return N # tamanho p+1, correspondendo aos índices (i-p .. i)
```

Com as funções básicas definidas, a próxima etapa será a definição do pseudocódigo para curva em NURBS.

5.1.3 Curvas

```
# Avalia ponto C(u) em uma curva NURBS
# Entradas: grau p, nós U, pontos P[0..n], pesos w[0..n], parâmetro u
function CurvePoint_NURBS(p, U, P, w, u):
  n = length(P) - 1
  i = FindSpan(n, p, u, U)
  N = BasisFuns(i, u, p, U)

  # Numerador e denominador racionais
  num = zero_vector_like(P[0]) # (x,y) ou (x,y,z)
  den = 0.0
```

```

for k in 0..p:
  idx = i - p + k
  Nk = N[k]
  wk = w[idx]
  wNk = wk * Nk
  num = num + (wNk * P[idx])
  den = den + wNk

if den == 0:
  raise "Denominador zero (verifique pesos e nós)."

C = num / den
return C

```

Com os códigos, pode-se seguir na definição do problema a ser resolvido para o estudo da análise Isogeométrica.

5.2 Viga de Bernoulli-Euler

Para simplificar a análise, utilizaremos o modelo da viga de Bernoulli-Euler, que consiste em uma simplificação da teoria linear da elasticidade, na qual definem hipóteses nas quais as seções planas permanecem planas após a deformação, seu deslocamento lateral da viga é nulo, e a deflexão vertical dos pontos dentro da seção transversal é pequena e igual às deformações do eixo da viga. Utilizaremos principalmente as ideias da viga de Euler, que consistem:

Instabilidade Elástica: a análise considera o comportamento de uma viga comprimida sujeita a um carregamento axial. Quando uma viga é suficientemente esbelta e submetida a uma força de compressão, ela pode falhar por instabilidade elástica antes de atingir seu limite de resistência.

Comprimento Efetivo da viga: é utilizado para levar em conta as condições de apoio e a forma como a viga é suportada em suas extremidades. Esse comprimento efetivo é usado para determinar a carga crítica de Euler, que é a carga máxima que a viga pode suportar antes de falhar por instabilidade elástica.

Equação de Euler: é uma equação diferencial de segunda ordem que descreve o comportamento de uma viga sujeita a uma força axial. Essa equação é usada para encontrar a carga crítica de Euler e determinar se a viga é estável ou instável sob compressão.

$$EI \frac{d^2y}{dx^2} + P(y - y_0) = 0$$

Onde:

- $y(x)$ é a função que descreve a deflexão da viga em relação à sua posição ao longo do eixo x ;
- E é o módulo de elasticidade do material da viga,
- I é o momento de inércia da seção transversal da viga,
- P é a carga axial aplicada,
- y_0 é a deflexão inicial da viga (se houver), e
- x é a posição ao longo do eixo axial da viga.

Neste contexto, considerando uma viga feita de material isotrópico e homogêneo, que possui comprimento L , e as cargas e momentos atuam no plano xz , que é o plano principal de inércia, o eixo x passa seu centro de gravidade G da seção transversal.

Chamando de “ u ” a deformação em x , “ v ” a deformação em y e “ w ” a deformação em z , tem-se o seguinte campo de deslocamento:

$$\begin{aligned}u(x, y, z) &= -z\theta(x) \\v(x, y, z) &= 0 \\w(x, y, z) &= w(x)\end{aligned}$$

Dado isso, baseando-se nas hipóteses das seções planas a rotação é igual a declividade do eixo da viga:

$$\begin{aligned}\theta &= \frac{dw}{dx} \\u &= -z \frac{dw}{dx}\end{aligned}$$

E a deformação específica é dada por:

$$\varepsilon_x = \frac{du}{dx}$$

Considerando que a viga é submetida a um estado de deformação axial pura, pela lei de Hook, tem-se:

$$\sigma_x = E\varepsilon_x = -zE \frac{d^2w}{dx^2}$$

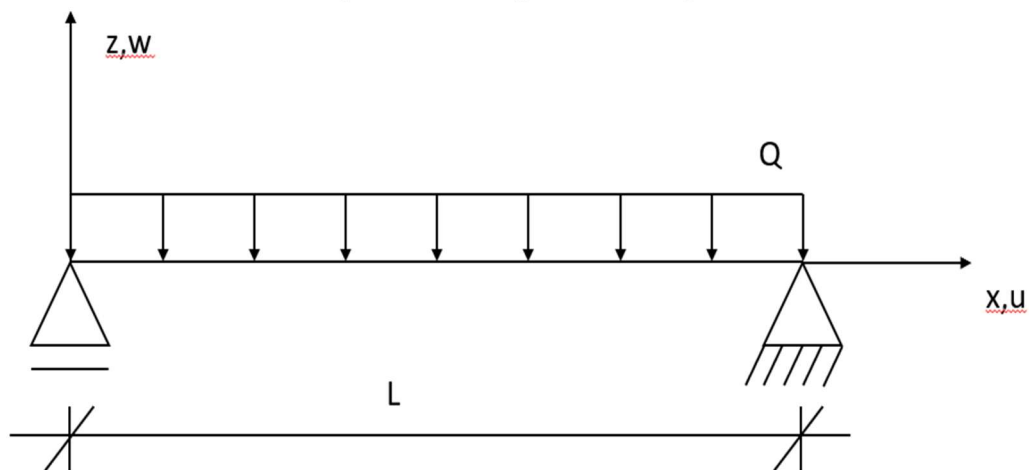
Portanto, a relação momento-curvatura é definida como:

$$M = - \iint z \sigma_x dA = \left(\iint_A z^2 dA \right) E \frac{d^2w}{dx^2} = EI_y \frac{d^2w}{dx^2} \quad (155)$$

Com isso, será adotada uma figura simples, bidimensional para a execução do programa, o que facilitará a chegada dos resultados através do modelo matemático,

sendo assim capaz de testar a precisão do método. Portanto, será analisada uma viga bi-apoiada isostática com um carregamento uniformemente distribuído, como apresentado na figura a seguir:

Figura 6 – Diagrama da viga



Fonte: Elaborado pelo autor, 2023

Com isso, consegue-se definir o *knot vector* para a execução da viga bi apoiada, seguindo aquelas regras de montagem nas quais já foram citadas anteriormente e assim montar o problema computacionalmente que seguirá a sequência de passos demonstrada pelo fluxograma de rotinas da figura 5.

Definindo os testes, tendo o *knot vector*, o grau de polinômio, P, o número de pontos de controle (NC) e quantidade de sub-regiões (NS), definido pela tabela seguir:

Tabela 1 – Aproximações para solução

Teste	Knot Vector	P	NC	NS
A	[0,0,0, 0.5,1,1,1]	2	4	2
B	[0,0,0,0,0.25,0.5,0.75,1,1,1,1]	3	7	4
C	[0,0,0,0,0,0.125,0.25,0.5,0.75,0.875,1,1,1,1,1]	4	10	6

Fonte: Elaborado pelo autor, 2023

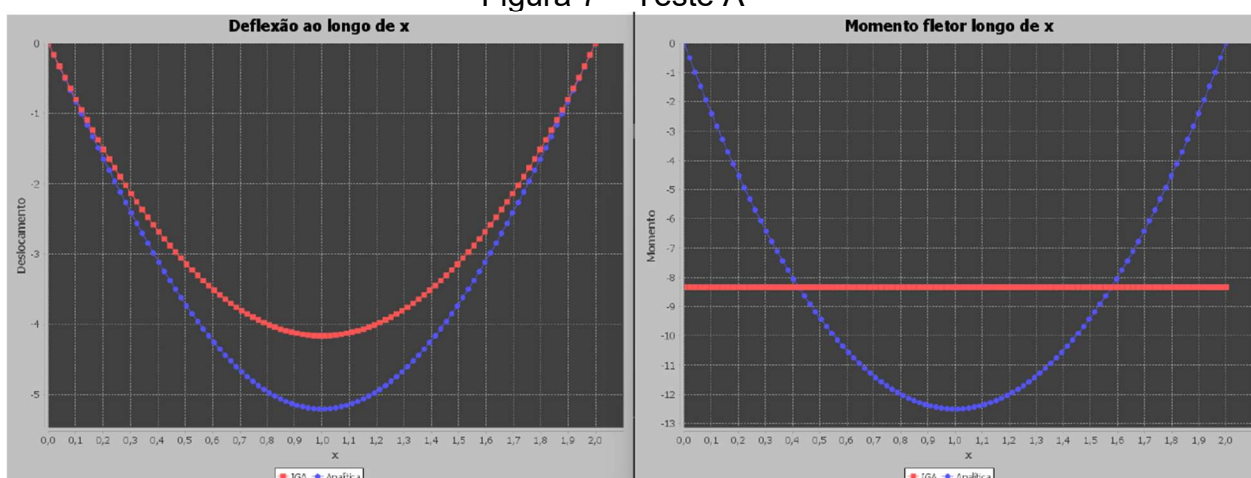
Além de análise será feita utilizando a equação de Von Mises, porque considera todas as componentes de tensão em um material e simplifica-as em uma única medida de tensão que é comparada com o limite de escoamento do material para determinar

se a deformação plástica irá ocorrer. Se a tensão de Von Mises exceder o limite de escoamento do material, é esperada uma deformação permanente.

6 RESULTADOS

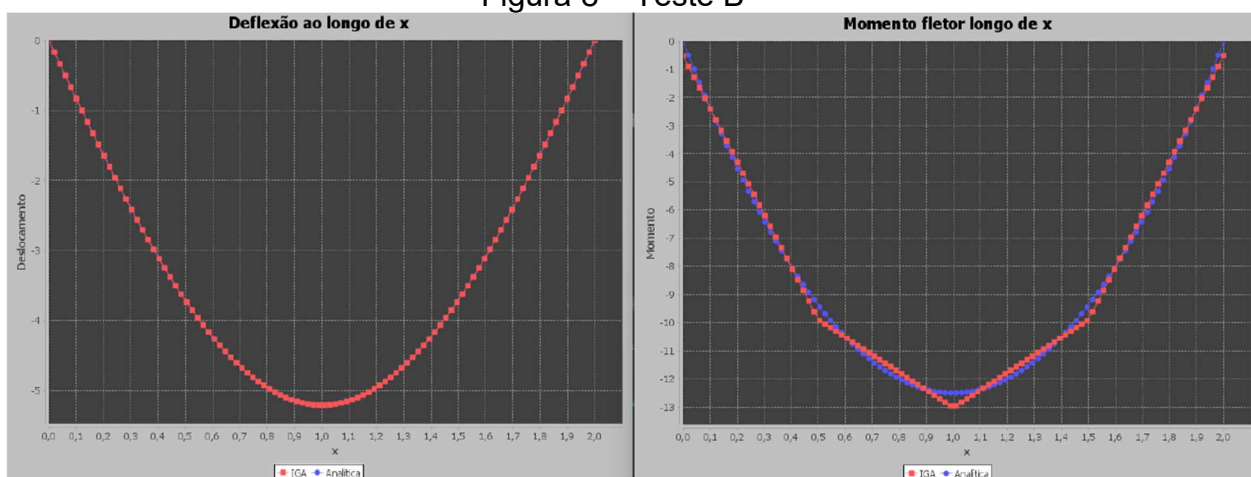
Estabelecida a metodologia, consegue-se chegar aos resultados da solução numérica e compará-los com os resultados que o programa em JAVA de análise Isogeométrica conseguiu chegar utilizando B-spline. Conforme na mostrada nas figuras a seguir, percebe-se a evolução da precisão em relação ao problema.

Figura 7 – Teste A



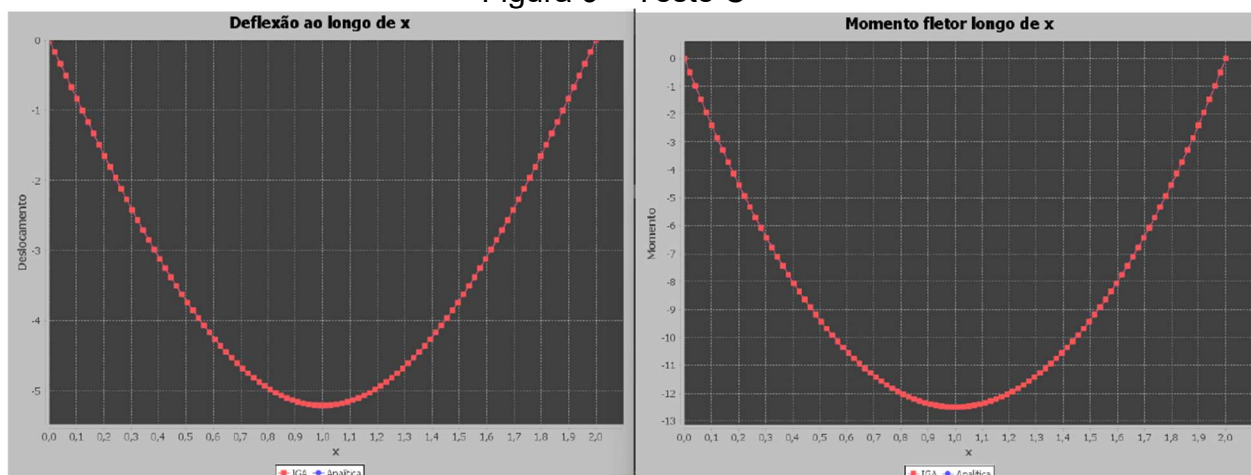
Fonte: Elaborado pelo autor, 2023

Figura 8 – Teste B



Fonte: Elaborado pelo autor, 2023

Figura 9 – Teste C



Fonte: Elaborado pelo autor, 2023

Mostrado a evolução, será comparado os resultados da deflexão e momentos máximos e os erros na tabela a seguir:

Tabela 2 – Resultados

TESTE	MÉTODO NUMÉRICO		ANÁLISE ISOGEOMÉTRICA		ERRO
	Deflexão	Momento	Deflexão	Momento	
A	5,208333	12,500000	4,166666	8,333333	0,179605
B	5,208333	12,500000	5,208333	13,020833	$7,02 \times 10^{-4}$
C	5,208333	12,500000	5,208333	12,499999	$3,29 \times 10^{-13}$

Fonte: Elaborado pelo autor, 2023

Conforme nota-se, em três testes conseguiu-se alcançar o resultado do modelo matemático, sendo que a deflexão, por sua simplicidade, foi alcançada já no segundo teste, com boa precisão e sem que houvesse tantos refinamentos, o que mostra a eficiência e funcionalidade da análise Isogeométrica para problemas de construção civil.

Além disso, os resultados mostram que o programa tem capacidade de ser aperfeiçoado para que seja testado em outros tipos de problemas da engenharia civil seja prosseguindo para diferentes tipos de vigas, pilares, treliças ou as estruturas nas quais ele apresenta grande vantagem sobre o MEF, cascas ou estruturas curvilíneas. Considerando a forma na qual foi estruturada o código, orientada para objetos, o aperfeiçoamento se torna mais simples, já que só há a necessidade de incluir novos módulos para isso.

7 CONCLUSÃO

Após a exposição da evolução dos métodos de cálculo estrutural computacional e a consolidação do Método dos Elementos Finitos (MEF) como a principal abordagem durante tantas décadas, evidencia-se a necessidade de explorar novos caminhos para aprimorar e executar sistemas amplamente utilizados, mas que ainda enfrentam desafios como o fenômeno do *locking* e as limitações na aproximação geométrica. Dessa forma, desenvolver uma ferramenta que se fundamente no MEF, porém integrando a geometria diretamente em sua concepção, revela-se não somente viável e interessante do ponto de vista computacional, mas também crucial para a inovação na análise estrutural.

Conforme demonstrado, a Análise Isogeométrica, proposta em 2005 pelos pesquisadores Hughes, Cottrell e Bazilevs, emergiu da identificação de problemas como a duplicidade na parametrização e a descontinuidade geométrica verificada nos métodos tradicionais de Elementos Finitos. O refinamento dos modelos paramétricos, aliado ao aprimoramento das ferramentas de CAD – inicialmente por meio de NURBS, com possibilidades de aplicação via B-Spline – possibilitou o desenvolvimento de uma metodologia mais otimizada para a resolução de problemas que envolvem contornos curvos e superfícies complexas.

A implementação do programa em JAVA, aliada à análise de um sistema simples de uma viga bi-apoiada, demonstrou que a abordagem proposta é altamente funcional para sistemas estruturais da engenharia civil, apresentando elevado grau de precisão com baixo custo computacional. Essa experiência piloto abre caminho para o desenvolvimento e testes de softwares mais robustos, aptos a realizar análises de estruturas com maior grau de complexidade.

Além disso, destaca-se que a Análise Isogeométrica, por ser um método interpolante que constrói a representação exata do problema, permite a definição direta das condições de contorno, resultando em vantagens como maior velocidade de processamento, redução de redundâncias nos cálculos e uma rápida convergência para resultados precisos. Tais características reforçam seu potencial para competir diretamente com o consagrado MEF.

Contudo, a difusão da Análise Isogeométrica ainda enfrenta barreiras: sua menor experimentação, o reconhecimento limitado e a tendência conservadora do ramo da construção civil, que privilegia soluções já estabelecidas e seguras. Ademais,

a transição para essa nova abordagem exige investimentos significativos em tempo, recursos humanos e financeiros – envolvendo treinamento de equipes, desenvolvimento ou aquisição de novos softwares e a adaptação dos processos existentes –, o que pode representar uma barreira para organizações de menor porte.

Portanto, é imperativo estimular o interesse e a pesquisa para problemas mais complexos, incentivando a produção acadêmica em língua portuguesa e ampliando a atuação de pesquisadores brasileiros na área. A injeção de recursos, tanto do setor público quanto privado, para o desenvolvimento de programas robustos e à altura dos consolidados métodos baseados no MEF, será fundamental para impulsionar a inovação e consolidar a Análise Isogeométrica como uma ferramenta de excelência na engenharia estrutural.

8 BIBLIOGRAFIA

A. Reali, **Isogeometric analysis approach for the study of structural vibrations**, *Journal of Earth Engineering*. Journal of Earthquake Engineering, 2006.

A. W. Wolfgang, F. Moritz, A. C. Cyron, **Isogeometric structural shape optimization**, *Computer Methods in Applied Mechanics and Engineering*. Technische Universität München, Alemanha, 2016.

B. A.M., B. M., B. Philipp, R. Wuchner, K. U. Bletzinger, **Nonlinear isogeometric spatial bernoulli beam**, *Computer Methods in Applied Mechanics and Engineering* **303**. Technische Universität München, Alemanha, 2016

BEER, Genot; BORDAS, Stéphane. **Isogeometric Methods For Numerical Simulation**. Itália: Springer, 2015.

BELLUOMINI, Nayra. **A evolução do CAD**. 2017. Disponível em: <https://blogs.autodesk.com/por-dentro-da-autodesk-brasil/2017/01/02/a-evolucao-do-cad/>. Acesso em: 22 jul. 2023

BESSA, André. **Java: o que é, linguagem e um Guia para iniciar na tecnologia**. Disponível em: https://www.alura.com.br/artigos/java?utm_term=&utm_campaign=%5BSearch%5D+%5BPerformance%5D. Acesso em: 24 jul. 2023.

CAZZANI, Antônio; MLAGÙ, Marcello; TURCO, Emilio. **Isogeometric Analysis of Plane Curve Beams**. Cagliari: MMS, 2014.

DEV MEDIA (org.). **Java: história e principais conceitos**. Disponível em: <https://www.devmedia.com.br/java-historia-e-principais-conceitos/25178>. Acesso em: 27 jul. 2023.

ENGINEER, Efficient (org.). **Understanding the Finite Element Method**. 2023. Disponível em: <https://efficientengineer.com/finite-element-method/>. Acesso em: 23 jul. 2023.

GAN, Buntara S. **An Isogeometric Approach to Beam Structures: Bridging the Classical to Modern Technique**. Koriyama, Fukushima, JP: Springer, 2018.

GAN, Buntara S. **Condensed Isogeometric Analysis for Plate and Shell Structures**. 1. ed. Nova York: CRC Press, 2019.

HUGHES, T. J. R.; COTTRELL, J. A.; BAZILEVS, Y. **Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement**. Austin, TX,

EUA: Institute for Computational Engineering and Sciences, The University of Texas at Austin, 2005.

MANNI, Carla; SPELEERS, Hendrik. **Geometric Challenges in Isogeometric Analysis**. Suíça: Springer, 2022.

PIEGL, Les; TILLER, Wayne. **The Nurbs Book**. Florida, EUA, 1995.

PEREIRA, H. C. R.; SANCHES, B. O.; PEREIRA, M. V. V.; THEODORO, D. L. V.; TORRES, W. A. R. **Análise Isogeométrica Aplicada em Problema de viga linear usando B-Splines**. IFMG, Santa Luzia, MG, Brasil, 2022

RAAMAC (org.). **Nurbs Fitting**. Disponível em:
<https://raamac.cee.illinois.edu/nurbs-fitting>. Acesso em: 22 jul. 2023.