

## **AN ISOGEOMETRIC BOUNDARY ELEMENT METHOD WITH FAST MULTIPOLE EXPANSION APPLIED TO PROBLEMS OF HEAT CONDUCTION**

**Emerson Bastos**

**Éder L. de Albuquerque**

*emersonbas@gmail.com*

*eder@unb.br*

*Mechanical Engineering Department, Technology College, University of Brasília*

*Asa Norte, Brasília, 70910900, Federal District (DF), Brazil*

**Lucas S. Campos**

*ufeslsc@gmail.com*

*Federal University of Espírito Santo*

*Goiabeiras, Vitória, 29075910, Espírito Santo, Brazil*

**Abstract.** This work presents an isogeometric formulation of the fast multipole boundary element method and its application in heat conduction problems. The formulation is developed using complex variables, expansion of fundamental solutions in Taylor series and using NURBS as shape functions. To reduce the computational cost and facilitate implementation, NURBS are decomposed into Bézier curves, making the isogeometric formulation more similar to the traditional boundary element method. Since influence matrices are not explicitly assembled, it is necessary to use an iterative method for solving the linear system. The generalized minimum residue method (GMRES) was chosen, based on previous work. A description of the hierarchical data structure and of the implemented algorithm is presented. Validation is performed by comparing results of the proposed formulation with those of the conventional boundary element formulation. The computational cost of both formulations are analyzed showing the advantages of the proposed formulation for large scale problems (problems with more than 100 thousand degrees of freedom).

**Keywords:** Boundary element method; Isogeometric analysis; NURBS; Fast multipole method; Iterative method.

## 1 Introdução

A análise isogeométrica foi introduzida por Hughes et al. [1] e descrita em detalhes por Cottrell et al. [2]. A idéia básica do conceito isogeométrico é usar as mesmas funções bases para a análise, como as que são usadas para descrever a geometria do problema, eliminando-se, como uma das suas principais vantagens, os erros de discretização que existem na formulação convencional do Método dos Elementos de Contorno (BEM).

Neste trabalho, a formulação isogeométrica estará munida de um recurso denominado operador extração de Bézier, conforme Borden et al. [3], que tem como objetivo extrair os elementos de Bézier isogeométricos para uso na análise isogeométrica baseada em funções splines racionais não-uniformes (NURBS). Este operador possibilita que a integração numérica seja realizada sobre elementos de Bézier, ou seja, fornece uma estrutura de elemento para a análise isogeométrica de equações integrais de contorno, muito similar aos elementos de contorno convencional.

A decomposição de Bézier é usada para calcular um conjunto de elementos de Bézier e com isso qualquer base pode ser usada desde que a partir dessa base se construa o operador extração de Bézier. E isso mostra-se ser de uma grande vantagem pois evita-se fazer uso da fórmula de recorrência e de alto custo computacional para o cálculo das funções bases das NURBS, Cox-de Boor, conforme Piegl and Tiller [4] e Rogers [5].

As quantidades físicas no contorno são interpoladas usando as funções bases NURBS, não-nulas e de suporte local, referentes ao segmento considerado na integração. Todavia, aplicar a formulação isogeométrica do método de elementos de contorno (IGABEM) diretamente, na forma convencional, consome muito tempo e memória quando se analisa problemas de larga escala (problemas com vários milhares de graus de liberdade), uma vez que sua ordem de complexidade de tempo é pelo menos  $O(N^2)$ , onde  $N$  é o número de graus de liberdade e as matrizes cheias e não simétricas do BEM se tornam muito grandes nestes casos. Para reduzir essa complexidade para  $O(N)$  ou  $O(N \log(N))$  como é mostrado por Nishimura and Liu [6], o Método Fast Multipole (FMM), cujo algoritmo é extensamente detalhado pelas referências Liu and Nishimura [7], Greengard and Rokhlin [8] e Rokhlin [9], é acoplado ao IGABEM, doravante denominado IGAFMBEM, e este, por sua vez, passa a ter capacidade de tratar problemas de larga escala.

Uma vez que o FMM dispensa o armazenamento da matriz dos coeficientes na memória do computador, faz-se necessário o uso de um solver iterativo e assim o Generalized Minimal Residual Method (GMRES), Saad and Schultz [10], é adotado de acordo com sua eficácia confirmada em trabalhos anteriores.

A proposta deste artigo é comparar a precisão e o custo computacional entre o algoritmo implementado da formulação IGAFMBEM e o BEM convencional usando elementos quadráticos contínuos.

## 2 As B-splines não uniformes racionais (NURBS)

As funções de forma lagrangianas usuais não são capazes de descrever geometrias mais complexas exatamente. Além disso, não preservam continuidade na derivada das variáveis, apresentando, assim, diversas fontes de erros nos cálculos. E é nesse contexto que entra em cena, efetivamente, as funções bases dos sistemas CAD/CAE, que são as funções não-uniformes racionais B-splines (NURBS). As NURBS são aptas na geração exata de geometrias complexas como as seções cônicas, ou seja, hipérbolas, elipses e parábolas, e esse foi um dos principais motivos de terem sido amplamente adotadas na indústria. Adicionalmente, também podem representar adequadamente as variáveis de campo.

### 2.1 Curva de Bézier

Pode-se definir uma curva de Bézier dependente do parâmetro real  $t$ , onde  $0 \leq t \leq 1$ , como:

$$\alpha(t) = \sum_{i=1}^{n+1} P_i J_i^n(t), \quad (1)$$

onde os coeficientes  $P_i$  são chamados pontos de controle que formam o polígono de controle e a base de Bernstein,  $J_i^n$ , é dada pelo polinômio de grau  $n$ :

$$J_i^n(t) = \binom{n}{i} t^i (1-t)^{n-i}, \quad (2)$$

$$\text{com } \binom{n}{i} = \frac{n!}{i!(n-i)!}.$$

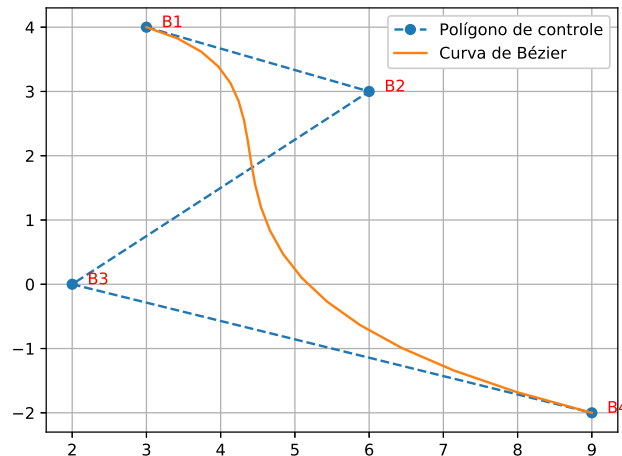


Figure 1. Curva de Bézier

Na Fig. 1, pode-se ver uma curva de Bézier de ordem  $n + 1 = 4$  e seu polígono de controle. Algumas propriedades importantes dessa curva são: (a) O número de pontos de controle menos um é o grau do polinômio gerador da curva; (b) A curva começa no primeiro ponto de controle e termina no último; (c) As inclinações dos vetores tangentes às extremidades desta curva, são as mesmas dos respectivos segmentos do polígono de controle que contém estas extremidades; (d) A curva está sempre contida na envoltória convexa dos pontos de controle; (e) A curva é invariante por transformação afim; (f) Cada ponto da curva é calculado como uma soma ponderada de todos os pontos de controle e desta forma, uma mudança de um ponto de controle afeta toda a curva globalmente.

## 2.2 Curva B-spline

As curvas B-splines são uma forma generalizada das curvas de Bézier. É composta de uma ou mais curvas de Bézier com um compromisso de continuidade entre as curvas/segmentos. Cada ponto de controle influencia apenas alguns segmentos da curva B-spline. Assim, pode-se ter um controle local da curva.

Uma curva B-spline é descrita por:

$$\alpha(t) = \sum_{i=1}^{n+1} P_i N_i^k(t), \quad t_{min} \leq t \leq t_{max}, \quad 2 \leq k \leq n+1, \quad (3)$$

onde  $P_i$  são os pontos de controle e  $N_i^k$  são as funções base de ordem  $k$  e grau  $k-1$ .

As funções  $N_i^k$  não são definidas explicitamente, mas são calculadas pela fórmula recursiva Cox-de Boor, Piegl and Tiller [4]:

$$N_i^1(t) = \begin{cases} 1, & \text{se } u_i \leq t < u_{i+1} \\ 0, & \text{se caso contrário} \end{cases} \quad (4)$$

e

$$N_i^k(t) = \frac{(t - u_i) N_i^{k-1}(t)}{u_{i+k-1} - u_i} + \frac{(u_{i+k} - t) N_{i+1}^{k-1}(t)}{u_{i+k} - u_{i+1}}, \quad (5)$$

onde  $u_i$ 's formam uma sequência não-decrescente de parâmetros reais compondo o vetor de knots  $U = \{u_0 \leq u_1 \leq u_2 \leq \dots \leq u_m\}$ .

O número de intervalos não-nulos no vetor  $U$  define o número de segmentos da curva B-spline. Cada segmento sofre influência de  $k$  pontos de controle. Cada vez que o parâmetro  $t$  entra em um novo intervalo, um novo ponto de controle se torna ativo, enquanto um ponto antigo é descartado. O número de knots,  $m$ , o número de pontos de controle,  $n+1$ , e a ordem da curva,  $k$ , estão relacionados por:

$$m = k + n + 1. \quad (6)$$

Pode-se repetir um knot consecutivamente no vetor  $U$  e ao número de vezes em que se repete dá-se o nome de multiplicidade do knot. Cada ponto de uma curva B-spline, de ordem  $k$ , é calculado como uma soma ponderada de  $k$  dos pontos de controle. Logo, uma mudança de um ponto de controle afetará a curva apenas localmente. Uma curva B-spline atende também as propriedades anteriores para curva de Bézier.

### 2.3 B-spline racional não uniforme

Em uma curva NURBS, um peso é associado à cada ponto de controle. Se esses pesos tiverem o mesmo valor, a curva será na realidade uma B-spline. As curvas NURBS são definidas como:

$$\alpha(t) = \frac{\sum_{i=1}^{n+1} P_i N_i^k(t) w_i}{\sum_{i=1}^{n+1} N_i^k(t) w_i} = \sum_{i=1}^{n+1} P_i R_i^k(t), \quad (7)$$

com

$$R_i^k(t) = \frac{N_i^k(t) w_i}{\sum_{i=1}^{n+1} N_i^k(t) w_i}, \quad (8)$$

onde  $R_i^k(t)$  é uma função base racional.

Pode-se notar que as curva NURBS são uma generalização da curva B-spline substituindo a base polinomial por uma razão de polinômios. Como observado no início desta seção, uma curva B-spline é um caso particular de uma curva NURBS e para mostrar isso basta tomar todos os pesos  $w_i^k$  tendo o mesmo valor na Eq. (7) e levar em conta que as funções bases B-spline possuem a propriedade da partição da unidade, conforme Piegl and Tiller [4] e Rogers [5]. Portanto, estas observações fazem com que suas características sejam basicamente as mesmas.

## 2.4 Inserção de Knot

Pode-se inserir knots em um vetor de knots sem alteração da geometria ou das propriedades da curva, porém, para cada knot inserido, um novo ponto de controle deve ser adicionado. Para que não haja alteração na continuidade das funções base  $R_i^k$ , esses novos knots,  $\bar{u}$ , e novos pontos de controle,  $\bar{P}_i$ , devem ser dados pelas fórmulas a seguir:

$$\bar{P}_i = \begin{cases} P_1, & \text{se } i = 1 \\ \alpha P_i + (1 - \alpha)P_{i-1}, & \text{se } 1 < i < r \\ P_n, & \text{se } i = r \end{cases}, \quad (9)$$

onde

$$\alpha = \begin{cases} 1, & \text{se } i = 1 \\ \frac{\bar{u} - u_i}{u_{i+k-1} - u_i}, & \text{se } 1 < i < r \\ 0, & \text{se } i = r \end{cases}. \quad (10)$$

Com  $r = (n + 1) + 1 = n + 2$ , os novos pontos de controle,  $\{\bar{P}_i\}_{i=1}^r$ , são formados a partir dos pontos de controle originais,  $\{P_i\}_{i=1}^{n+1}$ , conforme Borden et al. [3].

## 2.5 Decomposição de Bézier

A decomposição de Bézier é obtida inserindo knots repetidos em todos os knot internos ao vetor de knots  $U$ , até que eles tenham multiplicidade igual ao grau da curva. Após as inserções, a curva resultante é decomposta em um conjunto de elementos (curva) de Bézier, onde cada elemento corresponde a um intervalo de variação do vetor de knots  $U$ .

Considerando uma curva com  $n$  pontos de controle e chamando os  $r$  knots necessários para realizar uma decomposição, de  $\bar{u}_j$ , podemos definir  $\alpha_j$  conforme a Eq. (10). Pode-se, então, escrever uma matriz que relaciona os novos pontos de controle aos antigos:

$$C^j = \begin{bmatrix} \alpha_1 & 1 - \alpha_2 & 0 & 0 & \cdots & 0 \\ 0 & \alpha_2 & 1 - \alpha_3 & 0 & \cdots & 0 \\ 0 & 0 & \alpha_3 & 1 - \alpha_4 & \cdots & 0 \\ \vdots & & & & & \vdots \\ 0 & & \cdots & & \alpha_{n+j-1} & 1 - \alpha_{n+j} \end{bmatrix} \quad (11)$$

A Eq. (9) pode ser reescrita em forma matricial de maneira a representar a sequência de pontos de controle criadas pela inserção de knots, como:

$$\bar{P}^{j+1} = (C^j)^T \bar{P}^j . \tag{12}$$

Ao repetir essa operação  $r$  vezes, obtém-se a forma final da decomposição:

$$\bar{P} = C^T P , \tag{13}$$

onde  $C^T = (C^r)^T (C^{r-1})^T (C^{r-2})^T \dots (C^1)^T$ ,  $P$  é o conjunto original de pontos de controle e  $\bar{P}$  é o conjunto final, que pode ser chamado de pontos de controle de Bézier.

Como a inserção de knots não causa nenhuma alteração geométrica ou paramétrica, a curva de Bézier descrita pelos novos pontos de controle tem de coincidir com a spline anterior, assim:

$$\bar{P}^T J(t) = (C^T P)^T J(t) = P^T C J(t) = P^T N(t) , \tag{14}$$

onde  $J(t)$  e  $N(t)$  são, respectivamente, conjuntos de funções bases de Bernstein e de B-spline.

Logo, como  $P$  é arbitrário:

$$N(t) = C J(t) , \tag{15}$$

onde  $C$  é denominado operador extração de Bézier e depende apenas do vetor de knots  $U$ .

A seguir, apresenta-se um exemplo de uma curva NURBS cúbica na Fig. 2 e sua decomposição na Fig. 3.

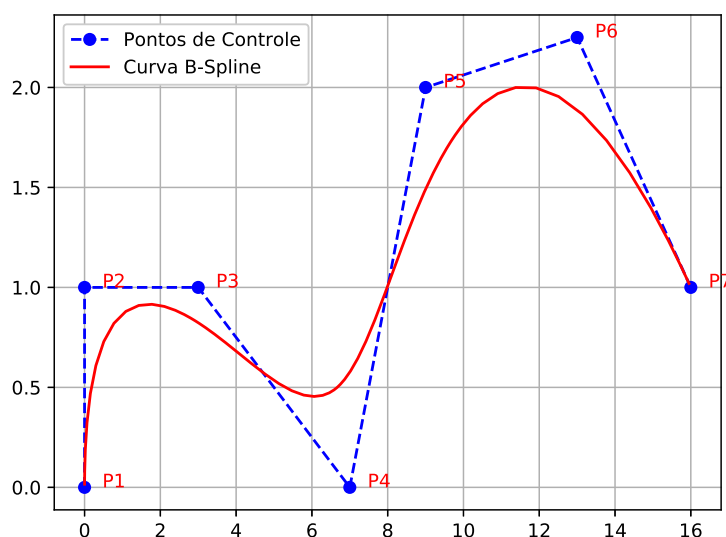


Figure 2. Uma curva NURBS e seus pontos de controle. O vetor de knot para a curva é  $\{0,0,0,0,1,2,3,4,4,4,4\}$ . P1, P2, P3, P4, P5, P6 e P7 são os pontos de controle.

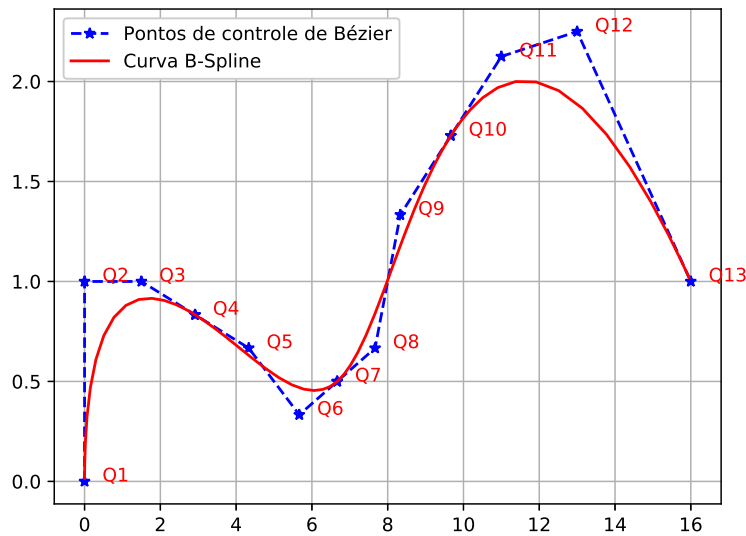


Figure 3. A decomposição da NURBS em curvas de Bézier.

Os polígonos de controle das curvas de Bézier na figura decomposta são Q1Q2Q3Q4, Q4Q5Q6Q7, Q7Q8Q9Q10 e Q10Q11Q12Q13. Daí, a integral de uma equação ao longo da curva da Fig. 2 pode ser realizada, de forma mais fácil e eficiente de um ponto de vista computacional, ao longo de cada um dos quatro elementos utilizando o operador extração de Bézier, conforme se verá na próxima seção.

### 3 Formulação do método dos elementos de contorno isogeométricos

A formulação clássica do método dos elementos de contorno, assim como todas as deduções algébricas das equações integrais podem ser encontradas em detalhes nas referências Brebbia [11], Brebbia et al. [12] e Brebbia and Dominguez [13]. Aqui será discutido como se realizará a integração da equação integral de contorno (BIE) nos elementos isogeométricos, com o contorno sendo visto como uma curva NURBS ou um "patch" de curvas NURBS, e que após a aplicação do operador extração de Bézier dado na Eq. (15), o contorno será discretizado em elementos de Bézier.

A aplicação do método proposto ocorrerá em um problema de condução de calor sem fonte interna, portanto o problema é descrito pela equação de Laplace, no domínio  $\Omega$  de contorno  $\Gamma$ , como:

$$\nabla^2 u(x, y) = 0, \quad (16)$$

onde  $x$  é um ponto fonte e  $y$ , um ponto campo.

Após as manipulações algébricas, obtem-se a BIE:

$$c(x)u(x) = \int_{\Gamma} \frac{\partial u(y)}{\partial n} u^*(x, y) d\Gamma - \int_{\Gamma} u(y) \frac{\partial u^*(x, y)}{\partial n} d\Gamma, \quad (17)$$

onde  $c(x)$  é um termo denominado "jump" que surge do processo de integração da equação integral e depende da geometria no ponto fonte  $x$ ,  $u$  e  $\frac{\partial u}{\partial n}$  são o potencial e sua derivada normal ao contorno  $\Gamma$ , enquanto que  $u^*$  e  $\frac{\partial u^*}{\partial n}$  referem-se às soluções fundamentais de potencial e de fluxo, respectivamente:

$$u^*(x, y) = -\frac{1}{2\pi} \log\left(\frac{1}{r}\right), \quad (18)$$

e

$$\frac{\partial u^*(x, y)}{\partial n(y)} = \frac{1}{2\pi r} \frac{\partial r}{\partial n}, \quad (19)$$

em que  $r$  é a distância entre o ponto fonte  $x$  e o ponto campo  $y$ .

A fim de discretizar a Eq. (17), deve-se escrever os campos contínuos  $u$  e  $\frac{\partial u}{\partial n}$  para cada elemento de Bézier, e isto conduz a:

$$c(x)u(x) = \sum_e \left( \int_{\Gamma_e} u^* \sum_{i=1}^{ne} \left( \frac{\partial u_i^c}{\partial n} R_i^k(t) \right) d\Gamma - \int_{\Gamma_e} \sum_{i=1}^{ne} \left( u_i^c R_i^k(t) \right) \frac{\partial u^*}{\partial n} d\Gamma \right). \quad (20)$$

E como dito antes, a formulação isogeométrica com o recurso da extração de Bézier simplifica bastante a definição de elemento, que resulta em:

$$c(x)u(x) = \sum_e \left( \sum_{i=1}^{ne} \left( \frac{\partial u_i^c}{\partial n} \int_{t_i}^{t_{i+k}} u^* R_i^k(t) \frac{d\Gamma}{dt} dt \right) - \sum_{i=1}^{ne} \left( u_i^c \int_{t_i}^{t_{i+k}} R_i^k(t) \frac{\partial u^*}{\partial n} \frac{d\Gamma}{dt} dt \right) \right), \quad (21)$$

onde, por sua vez, faz-se uma transformação para um domínio padrão possibilitando o uso da fórmula da quadratura de Gauss-Legendre:

$$\begin{aligned} & \sum_e \left( \sum_{i=1}^{ne} \left( u_i^c \left( \int_{-1}^1 R_i^k(t) \frac{\partial u^*(x, y)}{\partial n} \frac{d\Gamma}{dt} \frac{dt}{d\xi} d\xi + c(x) R_i^k(t) \right) \right) \right) \\ & = \sum_e \left( \sum_{i=1}^{ne} \left( \frac{\partial u_i^c}{\partial n} \int_{-1}^1 u^*(x, y) R_i^k(t) \frac{d\Gamma}{dt} \frac{dt}{d\xi} d\xi \right) \right), \end{aligned} \quad (22)$$

em que  $\frac{d\Gamma}{dt}$  é o Jacobiano da transformação do espaço coordenado físico para o espaço paramétrico, e  $\frac{dt}{d\xi}$  é o Jacobiano da transformação do espaço paramétrico para coordenada local.

Escrevendo a Eq. (22) matricialmente, obtemos:

$$Hu = Gq, \quad (23)$$

e rearranjando como no BEM convencional, chegamos ao sistema de equações lineares:

$$Ax = b, \quad (24)$$



### 3.1 Integração

Vale a pena enfatizar que na formulação isogeométrica do BEM, deve-se ter um cuidado especial no cálculo das integrais fortemente singulares, pois as mesmas não podem ser obtidas indiretamente através das hipóteses de temperatura constante ou movimento de corpo rígido, como é feito no BEM convencional.

Um método alternativo para lidar com singularidade forte, em casos 2D, é a técnica de subtração de singularidade apresentada por Guiggiani and et al. [14], mas neste trabalho, realiza-se a integração normalmente usando a quadratura de Gauss-Legendre padrão, pois, conforme Kane [15], mostra-se que há limitação de valores na integração do kernel dado na Eq. (19), na vizinhança do ponto de colocação. As integrais fracamente singulares são calculadas usando a transformação de Telles [16].

Os pontos de colocação estão distribuídos de acordo com os pontos da quadratura Gaussiana e independe da disposição das curvas de Bézier. No refinamento  $h$ , a cada inserção de um knot no vetor de knots,  $U$ , um ponto de controle e um ponto de colocação são adicionados aos conjuntos originais.

## 4 Aplicação do FMM ao IGABEM

Os métodos diretos necessitam de tempo e de memória com as ordens de complexidade computacional  $O(n^2)$  e  $O(n^3)$ , respectivamente, para resolver o sistema na Eq. (24), o que confirma a sua inviabilidade de uso em problemas de larga escala, de acordo com Liu [17]. Sendo assim, é necessário aplicar um método iterativo e como dito na introdução, o GMRES será utilizado dada sua eficácia comprovada em trabalhos prévios. E para isso teremos que calcular o produto matriz-vetor dado nos membros da Eq. (22), tarefa esta que será realizada pelo FMM.

Baseado na estrutura de árvore quaternária, dado um ponto fonte em uma célula folha, as integrais em elementos que se encontram nessa mesma célula ou em células vizinhas, chamadas integrais próximas, são calculadas diretamente, e as integrais em elementos que não se encontram em células vizinhas, chamadas integrais em elementos distantes, são calculadas pelo FMM, para mais detalhes veja Liu and Nishimura [7] e Liu [17].

O FMM utiliza operações denominadas translações, que são descritas brevemente aqui. Por conveniência, os pontos  $x$  e  $y$  são substituídos pela notação complexa:  $x = z_0 = x_1 + ix_2$  e  $y = z = y_1 + iy_2$ . Então

$$u^*(x, y) = Re\{u^*(z_0, z)\}, \quad (25)$$

onde

$$u^*(z_0, z) = -\frac{1}{2\pi} \ln(z_0 - z). \quad (26)$$

O kernel dado na Eq. (26) pode ser reescrito por expansão em série de Taylor, como segue:

$$u^*(z_0, z) = \frac{1}{2\pi} \sum_{\nu=0}^{\infty} I_{\nu}(z - z_c) O_{\nu}(z_0 - z_c), \quad (27)$$

onde

$$I_{\nu}(z - z_c) = \frac{(z - z_c)^{\nu}}{\nu!}, \quad \nu \geq 0. \quad (28)$$

$$O_\nu(z_0 - z_c) = \frac{(\nu - 1)!}{(z_0 - z_c)^\nu}, \quad \nu \geq 1. \quad (29)$$

$$O_0(z_0 - z_c) = -\ln(z_0 - z_c). \quad (30)$$

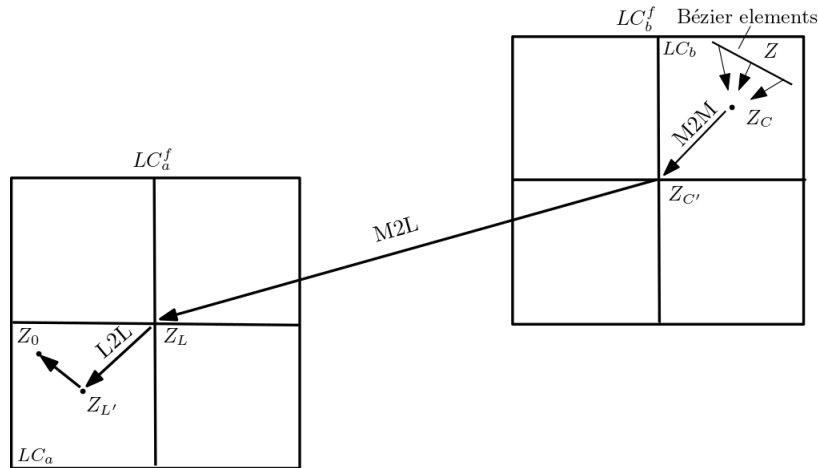


Figure 4. Translações no FMM

#### 4.1 Momentos multipolos

Os momentos multipolos são calculados em cada célula folha,  $LC_b$ , na estrutura da árvore, a partir dos elementos de Bézier naquela folha. Realativo à segunda parcela do segundo membro da Eq. (22), o momento multipolo pode ser escrito como

$$M_\nu(z_c) = \sum_{j=1}^{ne} \left( \int_{t_j}^{t_{j+k}} I_\nu(z - z_c) \left( \sum_{i=1}^{k+1} \frac{\partial u_i^c}{\partial n} R_i^k(t) \right) \frac{d\Gamma}{dt} dt \right). \quad (31)$$

E em relação à primeira parcela do segundo membro da Eq. (22), o momento multipolo pode ser escrito como

$$M_\nu(z_c) = \sum_{j=1}^{ne} \left( \int_{t_j}^{t_{j+k}} n(z) I_{\nu-1}(z - z_c) \left( \sum_{i=1}^{k+1} u_i^c R_i^k(t) \right) \frac{d\Gamma}{dt} dt \right), \quad (32)$$

onde  $z_c$  é o centro da folha,  $M_\nu(z_c)$  são os momentos multipolo em  $z_c$ ,  $n(z) = n_1 + in_2$  é o normal unitário exterior no ponto  $z$ ,  $k$  é a ordem do elemento de Bézier e  $ne$  é o número de elementos de Bézier na folha.

#### 4.2 Translação momento para momento (M2M)

O momento multipolo pode ser transformado da célula filha,  $LC_b$ , para a célula pai,  $LC_b^f$ , por translação M2M como

$$M_\nu(z_{c'}) = \sum_{m=0}^{\nu} I_{\nu-m}(z_c - z_{c'}) M_m(z_c), \quad (33)$$

onde  $z_{c'}$  é o centro da célula pai  $LC_b^f$ , veja Fig. 4.

### 4.3 Translação momento para local (M2L)

O momento multipolo pode ser transformado da célula  $LC_b^f$  para a célula  $LC_a^f$  por translação M2L como

$$L_l(z_L) = \frac{(-1)^l}{2\pi} \sum_{\nu=0}^{\infty} O_{l+\nu}(z_L - z_{c'}) M_\nu(z_{c'}), \quad (34)$$

onde  $z_{c'}$  é o centro da célula  $LC_b^f$  e  $z_L$  é o centro da célula  $LC_a^f$ , onde  $LC_b^f$  está na lista de interação de  $LC_a^f$ .  $L_l(z_L)$  é chamado coeficiente da expansão local.

### 4.4 Translação local para local (L2L)

O coeficiente de expansão local em uma célula, que não é uma folha, pode ser transformado para sua célula filha,  $LC_a$ , por translação L2L como

$$L_m(z_{L'}) = \sum_{l=m}^{\infty} L_l(z_L) I_{l-m}(z_{L'} - z_L), \quad (35)$$

onde  $z_{L'}$  é o centro da célula filha  $LC_a$ .

### 4.5 Integrais de elementos distantes

Finalmente, as integrais dos elementos distantes podem ser calculadas pelo FMM conforme a seguir:

$$I_{far} = \sum_{m=0}^{\infty} L_m(z_{L'}) I_m(z_0 - z_{L'}), \quad (36)$$

onde  $I_{far}$  é o valor das integrais distantes na Eq. (22).

## 5 Resultados

Nesta seção, com objetivo de avaliar o potencial da formulação isogeométrica proposta, um exemplo numérico em duas dimensões é analisado. Os resultados são comparados com BEM convencional usando elementos quadráticos contínuos e a solução analítica do exemplo.

O processamento foi realizado em um computador notebook Acer, com processador Intel i5 (2.3 GHz com Turbo Boost até 2.8 GHz), 8 Gigabytes de memória RAM. A linguagem de programação utilizada foi Julia, na versão 1.0.3.

## 5.1 Configuração numérica

Nesta formulação proposta, será utilizado o refinamento  $h$ , que significa inserção de knots no vetor de knots  $U$ , conforme Piegl and Tiller [4] e Cottrell et al. [2]. E deve ser observado que aumentando-se o valor de  $h$ , conseqüentemente, aumenta o número de graus de liberdade, mas com algum compromisso deve-se aumentar, também, o número de pontos de Gauss-Legendre para não haver diminuição da precisão. O motivo desse compromisso ainda será objeto de investigação futura.

Este estudo emprega o GMRES, sem condicionamento, para resolver o sistema na Eq. (24). Os parâmetros principais são os seguintes:

- Tolerância (tol) = 1e-6;
- Número de truncamento da expansão multipolo = 30
- Número de truncamento da expansão local = 30
- Número máximo de elementos em uma folha = 10

O critério de parada usual consiste na ponderação, pela tolerância, do resíduo relativo ( $R_e$ )

$$Re^{(i)} = \frac{|Ax^{(i)} - b|_2}{|b|_2}, \quad (37)$$

onde  $x^{(i)}$  representa um candidato à solução do sistema da Eq. (24) no  $i$ -ésimo passo e  $|\cdot|_2$  denota a norma  $l^2$ .

O instrumento que será utilizado para medir a precisão de cada método e comparar um com o outro, será a raiz quadrada do erro quadrático médio (RMS) não normalizado:

$$error_{RMS} = \sqrt{\frac{\sum_{t=1}^n (y_t - y_a)^2}{n}}, \quad (38)$$

onde  $y_t$  é o valor numérico obtido no ponto  $t$ ,  $y_a$  é o valor analítico,  $n$  é o número de pontos de colocação.

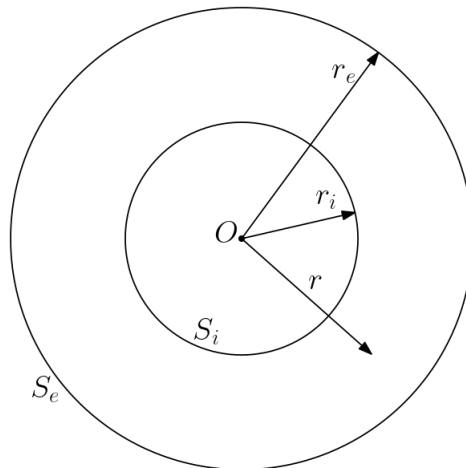


Figure 5. Problema potencial em uma região anular

Foi realizado diversas simulações com o número de truncamento da expansão local variando em 15, 20, 25, 30, 35 e 40, daí observou-se que o valor de melhor ajuste entre tempo de convergência, pelo GMRES, e precisão foi 30. O número de pontos de Gauss-Legendre foi fixado em 36 de acordo, também, com o critério de ajuste entre tempo de convergência e precisão. Outro parâmetro de sensível influência no desempenho do IGAFMBEM e que teve seu valor escolhido dentre diversos foi o número máximo de elementos numa folha, aqui escolhido como 10. Notou-se que quanto maior era seu valor, maior era o tempo de execução do produto matriz-vetor, dado pela Eq. (21), e de convergência pelo GMRES.

## 5.2 Transferência de calor em um cilindro vazado

O problema a ser estudado é o de transferência de calor em um cilindro vazado, como mostrado na Fig. 5. Este problema tem uma solução analítica, que será usada para confirmar os resultados. As condições de contorno conhecidas são a temperatura no contorno interno,  $S_i$ , e fluxo no contorno externo,  $S_e$ .

A solução analítica para a temperatura é dada por:

$$T(r) = T_i - q_e r_e \log\left(\frac{r}{r_i}\right), \quad (39)$$

e para o fluxo:

$$q(r) = -q_e \frac{r_e}{r}, \quad (40)$$

onde  $T_i$  e  $q_e$  são a temperatura e o fluxo nos contornos interno e externo, respectivamente.

Os valores considerados são:  $r_i = 1$ ,  $r_e = 2$ ,  $T_i = 100$ ,  $q_e = -200$  e a condutividade térmica tendo valor constante 1. Os valores analíticos para esses dados são  $T_e = 377.2588722$  e  $q_i = 400$ .

Nas Figs. 6 e 7, pode-se observar o erro RMS para a temperatura ao longo dos contornos  $S_i$  e  $S_e$ . Observa-se que os dois métodos convergem para a solução exata, porém o método isogeométrico com muito menos graus de liberdade atinge a mesma precisão que o método convencional e isto pode ser constatado também na Tabela 1. Com 640 graus de liberdade, o método convencional obtém uma precisão equivalente à precisão obtida pelo método proposto usando apenas 48 graus de liberdade.

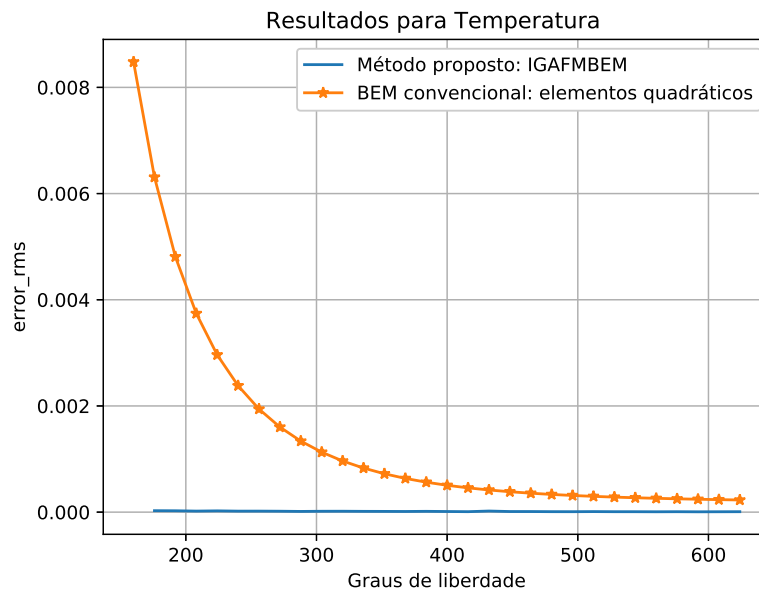


Figure 6. Erro RMS da temperatura nos contornos  $S_i$  e  $S_e$ .

Um cenário de superioridade, ainda maior, na performance do método proposto, sobre o método convencional, é verificado para o cálculo do fluxo, conforme nota-se nas Figs. 8 e 9, e também pela Tabela 2. Aqui novamente com 640 graus de liberdade, o método convencional obtém uma precisão que é inferior, cerca de 37 vezes, à precisão obtida pelo método proposto usando apenas 48 graus de liberdade. Diferença muito grande também se pode notar nas demais linhas da Tabela 2.

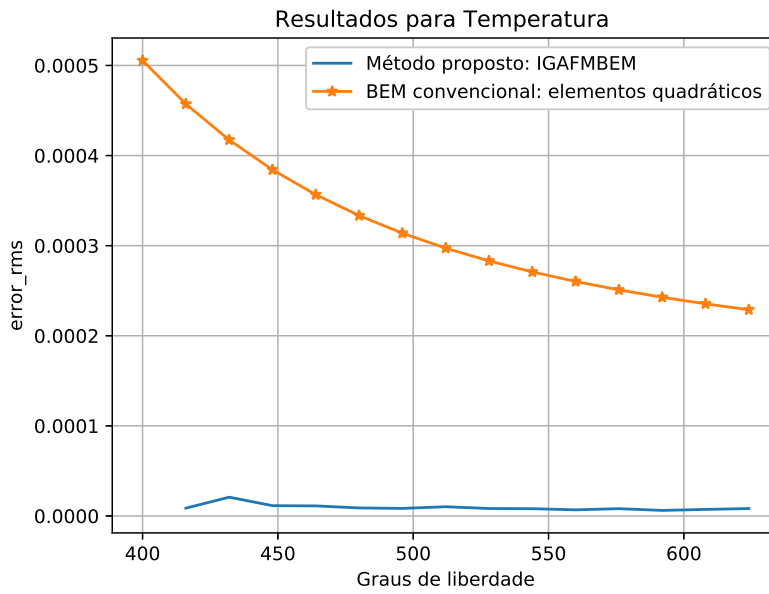


Figure 7. Erro RMS da temperatura nos contornos  $S_i$  e  $S_e$ .

Table 1. Erro RMS da temperatura

Elementos quadráticos		Elementos isogeométricos de grau 2	
DOF	Erro RMS	DOF	Erro RMS
48 (640)	3.17e-1 (2.29e-4)	48	1.997e-4
96	4.025e-2	96	3.38e-5
176	6.31e-3	176	2.71e-5

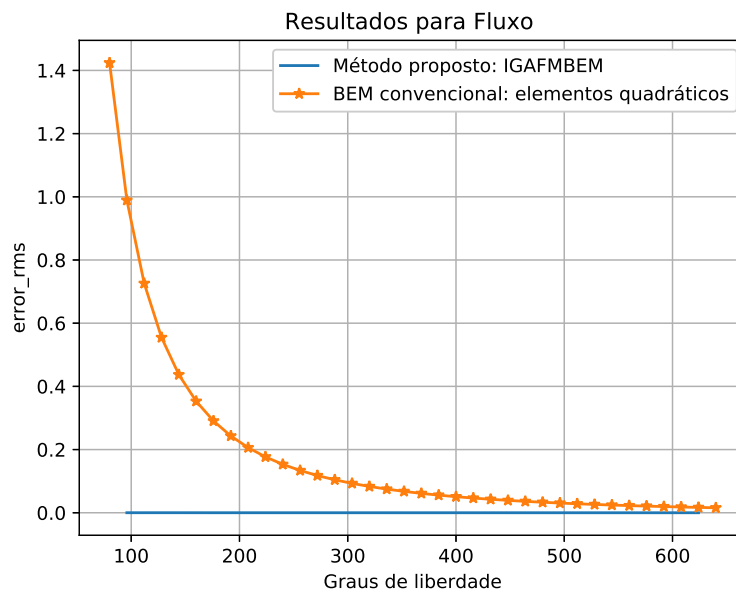


Figure 8. Erro RMS do fluxo nos contornos  $S_i$  and  $S_e$ .

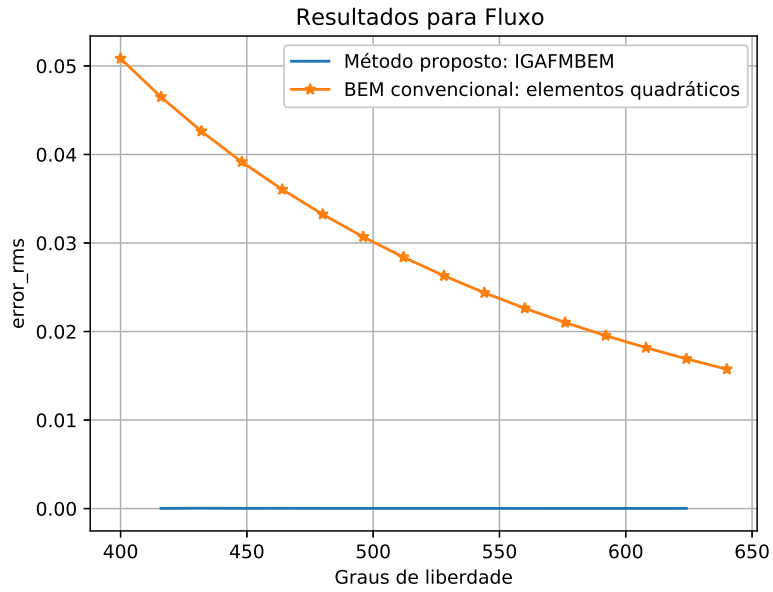


Figure 9. Erro RMS do fluxo nos contornos  $S_i$  and  $S_e$ .

Table 2. Erro RMS do fluxo

Elementos quadráticos		Elementos isométricos de grau 2	
DOF	Erro RMS	DOF	Erro RMS
48 (640)	8.48 (1.57e-2)	48	4.26e-4
96	9.89e-1	96	9.92e-5
176	2.91e-1	176	5.96e-5

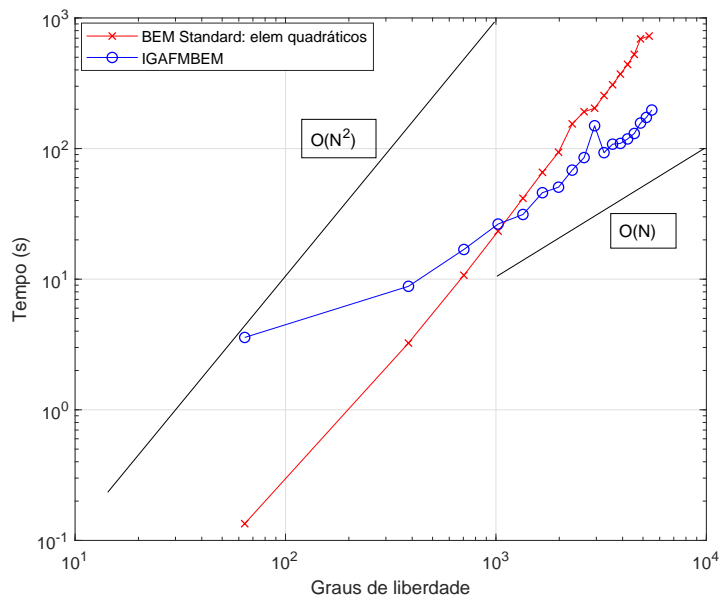


Figure 10. Tempo de execução: IGAFMBEM vs BEM convencional.

O tempo de execução para o número crescente de graus de liberdade é mostrado na Fig. 10. A mudança na inclinação das curvas indicam a melhoria na velocidade para problemas de larga escala. Pode-se notar a complexidade computacional  $O(N^2)$  para o método convencional e  $O(N)$  para o IGAFMBEM. Estes resultados estão de acordo com as referências Matsumoto and Takahashi [18] e Wang et al. [19].

Neste trabalho preferiu-se comparar o método proposto com o método convencional utilizando elementos quadráticos contínuos, pois a comparação deste com o elementos constantes pode ser verificado nas referências Braga [20], Campos [21] e Campos et al. [22]. E lá constata-se desempenho melhor dos elementos quadráticos sobre os elementos constantes.

## 6 Conclusões

Este trabalho faz a acoplagem entre a formulação isogeométrica (NURBS) e o método Fast Multipole, promovendo a discretização exata da geometria a partir dos parâmetros do CAD, ganhando velocidade na computação e tornando o método capaz de lidar com problemas de larga escala. Como consequência, tem-se economia no uso de memória, uma vez que será usado o método iterativo GMRES para resolução do sistema linear, e assim as matrizes cheias e não simétricas do BEM convencional não precisam ser armazenadas.

As condições de contorno são impostas muito facilmente, dada a forma de distribuição no posicionamento dos pontos de colocação, onde fica garantido que não haverá nenhum destes em quinas, se por ventura existirem. A facilidade da imposição das condições de contorno são devidas ao uso dos vetores de conectividade entre os elementos/curvas de Bézier.

O refinamento  $p-h$  é viabilizado pelo recurso do operador extração de Bézier, que por sua vez torna o método proposto muito semelhante ao BEM convencional, facilitando sobremaneira a implementação.

A ordem de complexidade computacional para o tempo de execução do método é reduzida de  $O(n^2)$  para  $O(n)$ . Portanto, ganha-se mais rapidez, usa-se menos memória e tem-se mais precisão, comparando-se com métodos convencionais de elementos de contorno.

## References

- [1] Hughes, T. J. R., Cottrell, J. A., & Bazilevs, Y., 2005. Isogeometric analysis: Cad, finite elements, nurbs, exact geometry, and mesh refinement. *Computer Methods in Applied Mechanics and Engineering*, vol. 194, pp. 4135–4195.
- [2] Cottrell, J. A., Hughes, T. J. R., & Y., B., 2009. *Isogeometric Analysis: Toward Integration of CAD and FEA*. Wiley: Chichester, U.K.
- [3] Borden, M. J., Scott, M. A., Evans, J. A., & Hughes, T. J. R., 2011. Isogeometric finite element data structures based on bézier extraction of nurbs. *International Journal for Numerical Methods in Engineering*, vol. 87, pp. 15–47.
- [4] Piegl, L. & Tiller, W., 1997. *The NURBS Book (Monographs in Visual Communication) (2nd edn)*. Springer: New York.
- [5] Rogers, D. F., 2001. *An Introduction to NURBS with Historical Perspective*. Academic Press: San Diego, CA.
- [6] Nishimura, N. & Liu, Y., 2002. The fastmultipole accelerate boundary integral equation methods. *Appl Mech Rev*, vol. 55(4(July)), pp. 299–324.
- [7] Liu, Y. & Nishimura, N., 2006. The fast multipole boundary element method for potential problems: A tutorial. *Engineering Analysis With Boundary Elements*, vol. 30(5), pp. 371–381.



- [8] Greengard, L. & Rokhlin, V., 1987. A fast algorithm for particle simulations. *Journal of computational physics*, vol. 73(2), pp. 325–348.
- [9] Rokhlin, V., 1985. Rapid solution of integral equations of classical potential theory. *Journal of Computational Physics*, vol. Elsevier, v. 60, n. 2, pp. 187–207.
- [10] Saad, Y. & Schultz, M. H., 1986. Gmres: a generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM J Sci Stat Comput*, vol. 7(3), pp. 856–869.
- [11] Brebbia, C. A., 1978. *The boundary Element Method for Engineers*. Pentech Press, London.
- [12] Brebbia, C. A., Wrobel, L. C., & Telles, J. C. F., 1984. *Boundary Element Techniques: Theory and Applications in Engineering*. [S.I.]: WIT Press.
- [13] Brebbia, C. A. & Dominguez, J., 1989. *Boundary elements: an introductory course*. [S.I.]: WIT Press.
- [14] Guiggiani, M. & et al., 1992. A general algorithm for the numerical solution of hypersingular boundary integral equations. *Journal of applied mechanics, American Society of Mechanical Engineers*, vol. v. 59, n. 3, pp. 604–614.
- [15] Kane, J. H., 1994. *Boundary element analysis in engineering continuum mechanics*. Englewood Cliffs, NJ: Prentice Hall.
- [16] Telles, J. C. F., 1987. A self-adaptive co-ordinate transformation for efficient numerical evaluation of general boundary element integrals. *International Journal for Numerical Methods in Engineering*, vol. v. 24, n. 5, pp. 959–973.
- [17] Liu, Y. J., 2009. *The fastmultipole boundary element method*. Prentice-Hall.
- [18] Matsumoto, T. & Takahashi, T., 2012. An application of fast multipole method to isogeometric boundary element method for laplace equation in two dimensions. *Engineering analysis with boundary elements*, vol. 36, pp. 1766–1775.
- [19] Wang, Q., Zhou, W., Cheng, Y., Ma, G., Chang, X., & Liu, B., 2019. A nurbs-enhanced improved interpolating boundary element-free method for 2d potential problems and accelerated by fast multipole method. *Engineering analysis with boundary elements*, vol. 98, pp. 126–136.
- [20] Braga, L. M., 2012. *O método dos elementos de contorno rápido com expansão em multipolos aplicado a problemas de condução de calor*. PhD thesis, UnB, Brasília, DF, Brazil.
- [21] Campos, L. S., 2016. *Método dos elementos de contorno isogeométricos acelerado pela aproximação cruzada adaptativa*. PhD thesis, UnB, Brasília, DF, Brazil.
- [22] Campos, L. S., Albuquerque, E. L., & Wrobel, L. C., 2017. An aca accelerated isogeometric boundary element analysis of potential problems with non-uniform boundary conditions. *Engineering Analysis with Boundary Elements*, vol. 80, pp. 108–115.