

Construção de Superfícies Suaves com Continuidade Geométrica

F.C. DA ROCHA^{1*} e H.B. CODA²

Recebido em 7 de julho de 2017 / Aceito em 10 de março de 2018

RESUMO. No uso de métodos numéricos, principalmente o método dos elementos de contorno, a construção de superfícies suaves é um dos requisitos importantes para a modelagem computacional de sólidos. Adicionalmente aos métodos numéricos, existem muitos outros campos de aplicação, tais como, na animação computacional e na reconstrução de imagens médicas, em que a continuidade geométrica G^1 é condição fundamental para a geração de superfícies suaves. No presente trabalho é utilizada a técnica de construção de geometria a partir das curvas do contorno. Diferentemente das construções clássicas das funções aproximadoras C^0 , a presente construção G^1 faz uso dos vetores normais nos vértices da malha triangular, além de suas coordenadas geométricas. No entanto, os dados de entrada, especialmente os vetores normais, não são de obtenção imediata para geométricas complexas. Com o intuito de contornar esta dificuldade, o presente trabalho desenvolveu um código computacional para construção de superfícies suaves acoplado com o software de computação gráfica *BlenderTM*. Desta forma, a implementação computacional tanto para a criação dos elementos triangulares com continuidade geométrica quanto para a obtenção das informações iniciais, via *BlenderTM*, é explicitada e sua eficiência é comprovada por meio de modelagem de geometrias globalmente paramétricas e não-paramétricas de complexidade variada.

Palavras-chave: Continuidade G^1 , elemento triangular, software *BlenderTM*, geometria complexa.

1 INTRODUÇÃO

A construção de superfícies suaves é uma das principais exigências na modelagem das mais diversas geometrias, que vão desde a carenagem de automóveis ou fuselagem de aviões a objetos presentes em filmes de animação ou jogos interativos.

Um algoritmo para interpolação suave de dados 3D usando superfície composta é obtida a partir da subdivisão e subsequente correção de cada elemento, tanto dentro quanto através dos contornos [7]. Isso significa que uma correção é executada para cada lado na triangularização, o que não é desejável quando as superfícies são ajustadas a um grande conjunto de dados de entrada. O método apresentado em Farin [7] é modificado por Piper [17] para considerar o vetor

*Autor correspondente: Fabio Carlos da Rocha – E-mail: fcrocha@ufs.br

¹Departamento de Engenharia Civil, DEC, Universidade Federal de Sergipe, Av. Marechal Random, s/n, 494000-000, São Cristóvão, SE, Brasil.

²Departamento de Engenharia de estruturas, EESC, Universidade de São Paulo, Av. Trabalhador São-carlense, 400, 13566-590, São Carlos, SP, Brasil. E-mail: hbcoda@sc.usp.br

tangente como dado de entrada, assim como, novos valores para as funções pesos λ , μ e ν . No entanto, ainda é necessário proceder as correções para cada lado na triangularização. Com o objetivo de construir superfícies suaves e evitar as correções no processo de triangularização, são construídas superfícies cúbicas G^1 aproximadas, entretanto, à custa de sucessivos refinamentos da triangularização [18].

Dando continuidade, elementos triangulares que possuem, na interseção, o mesmo plano tangente é objetivo de desenvolvimento usando curvas de contorno [21, 14]. As curvas de contorno adotadas são inicialmente cúbicas e então são elevadas para ordem quatro a partir da obtenção de pontos internos adicionais aos, já existentes, pontos de controle da curva cúbica por meio da técnica de interpolação ponto-normal [20]. Na tentativa de estimar o vetor normal, autores propuseram a utilização da média normalizada dos vetores normais aos triângulos que contém o vértice em comum [21]. E outros propuseram estimar, este vetor, a partir da média ponderada das normais em um mesmo vértice [18]. Nesta média, os pesos são proporcionais às áreas dos triângulos ou aos ângulos que se encontram nos vértices.

Dentre as novas propostas de construção de superfícies suaves utilizando interpolação ponto-normal, tem-se a construção de elementos triangulares curvos a partir da interpolação polinomial cúbica de Bézier cujos pontos de controle são obtidos por meio das projeções dos planos tangentes nos vértices [19], e pela utilização de um fator de forma para regular a planicidade da superfície [4]. Outros trabalhos propõem a substituição em cada borda do triângulo plano por uma curva ortogonal às normais dadas nos vértice, e então aproximando o interior do elemento triangular por um polinômio quadrático de Bézier que satisfaça às condições das curvas do contorno do triângulo [13]. Ou utilizando a aproximação dos mínimos quadrados para ajusta a superfície quadrática interna às curvas de contorno [1].

No presente trabalho é apresentado um procedimento de construção de superfícies suaves a partir da interpolação ponto-normal. Neste procedimento, os dados de entrada, coordenadas geométricas e os vetores normais nos vértices dos triângulos da malha, são obtidos com auxílio do software de computação gráfica *BlenderTm* que é acoplado ao programa de geração de superfícies suaves desenvolvido em FORTRAN 11.0. Em seguida, os resultados obtidos pela presente proposta são comparados com os obtidos pela formulação PN [19], Phong Tessellation [4], Nagata [13] e NLSA [1].

Este trabalho está organizado da seguinte forma. No item 2 é apresentada a formulação para criação de superfícies suaves. No item 3 é apresentado o procedimento para construção de elementos G^1 . No item 4, o código fonte para obtenção de vetores normais via *BlenderTm* 2.70 é apresentado. Em seguida, no item 5, são apresentados exemplos de aplicação. Por fim, no item 6 são realizadas as considerações finais.

2 FORMULAÇÃO PARA CRIAÇÃO DE SUPERFÍCIES SUAVES

Nesta seção são apresentados os fundamentos básicos necessários para a construção de superfícies, assim como é mostrada a formulação teórica dos elementos triangulares com continuidade geométrica G^1 .

2.1 Superfícies de Bézier

A abordagem mais elegante e matematicamente mais natural de construir elementos triangulares é por meio das coordenadas baricêntricas. Define-se as coordenadas baricêntricas (u, v, w) de um ponto $\mathbf{p} = (x, y)$, com respeito a um triângulo de referência T , pela razão entre áreas

$$u = \frac{\Delta_1}{\Delta}, v = \frac{\Delta_2}{\Delta}, w = \frac{\Delta_3}{\Delta}, \quad (2.1)$$

de modo que $\mathbf{p} = u\mathbf{p}_1 + v\mathbf{p}_2 + w\mathbf{p}_3$, $\mathbf{p}_k = (x_k, y_k)$ e $u + v + w = 1$. Na equação (2.1), tem-se que Δ representa a área do triângulo de referência T , e Δ_k , com $k = 1, 2, 3$, representam as áreas dos sub-triângulos do triângulo de referência, que são formados por dois vértices e um terceiro ponto interno de coordenada (x, y) .

A partir da expansão de $1 = (u + v + w)^n$ é obtida, em duas variáveis, a base de Bernstein de grau n definida para $0 \leq i, j, k \leq n$ com $i + j + k = n$, por [12]

$$B_{ijk}^n(u, v, w) = \frac{n!}{i!j!k!} u^i v^j w^k. \quad (2.2)$$

Existem $\frac{1}{2}(n+1)(n+2)$ funções base linearmente independentes. Desta forma é obtido um mapeamento do domínio paramétrico T para \mathbb{R}^3 por meio do elemento triangular de Bézier-Bernstein definido por [2]

$$\mathbf{S}(u, v, w) = \sum_{|\mathbf{i}|} \mathbf{p}_{i,j,k} B_{ijk}^n(u, v, w) \quad 0 \leq i, j, k \leq n, \quad (2.3)$$

em que $\mathbf{p}_{i,j,k}$ são pontos de controle e $|\mathbf{i}| = i + j + k = n$.

2.2 Continuidade geométrica

A condição de continuidade de função (assim como a de coplanaridade para o caso de superfícies) não implica em continuidade dos planos tangentes nas interseções destas funções. Desta forma, a garantia da existência de suavidade na interseção entre elementos (funções) está associada a continuidade geométrica, a qual está intrinsecamente relacionada ao conceito de parametrização equivalente.

Seja $C(u)$, com $u \in [a, b]$, e $\tilde{C}(\tilde{u})$ com $\tilde{u} \in [\tilde{a}, \tilde{b}]$, duas parametrizações regulares C^∞ (uma parametrização é regular se sua primeira derivada existe). Essas parametrizações são ditas equivalentes, isto é, descrevem a mesma curva orientada, se existe uma função $C^\infty f: [\tilde{a}, \tilde{b}] \rightarrow [a, b]$ tal que [12]

1. $\tilde{C}(\tilde{u}) = C(f(\tilde{u}))$;
2. $f(\tilde{a}) = a$;
3. $f(\tilde{b}) = b$;
4. $\frac{df(\tilde{u})}{d\tilde{u}} > 0$

A definição de parametrização equivalente para superfícies pode ser obtida de forma semelhante à definição para curvas, apenas acrescentando um novo parâmetro \tilde{v} e considerando a função de mudança, f , com dois parâmetros em seu domínio.

A partir do conceito de parametrização equivalente, define-se a continuidade geométrica: duas superfícies regulares $\mathbf{S}_1(s, t)$ e $\mathbf{S}_2(u, v)$ se interceptam com continuidade G^k no ponto $\mathbf{P} = \mathbf{S}_1(s_0, t_0) = \mathbf{S}_2(u_0, v_0)$ sempre que existir um mapeamento invertível (chamado de reparametrização) $f: [\tilde{u}, \tilde{v}] \rightarrow [u(\tilde{u}, \tilde{v}), v(\tilde{u}, \tilde{v})]$ que proporcione $\mathbf{S}_1(s, t)$ e $\mathbf{S}_2(u(\tilde{u}, \tilde{v}), v(\tilde{u}, \tilde{v}))$ se interceptarem com continuidade C^k neste ponto. Assim, duas superfícies são consideradas de continuidade G^k se a interseção entre as superfícies tem continuidade G^k em cada ponto da curva comum às superfícies.

Portanto, quando duas superfícies se interceptam com continuidade G^1 em todo ponto \mathbf{P} , as superfícies apresentam normais proporcionais e, como consequência, o mesmo plano tangente na região de interseção das superfícies. O inverso também é verdadeiro, se duas superfícies se interceptam no ponto \mathbf{P} e tem normais às superfícies proporcionais neste ponto, então as superfícies se interceptam com continuidade G^1 [11].

2.3 Técnica para construção de elementos G^1

A ideia chave do presente algoritmo é transformar malhas inicialmente constituída por triângulos planos em malhas formadas por triângulos curvos, ou seja, o vetor posição (ou coordenada geométrica) \mathbf{Q}_i ($\mathbf{Q}_i = \mathbf{p}_i$) e o vetor normal \mathbf{N}_i nos vértices dos triângulos planos são interpolados pela superfície paramétrica triangular de Bézier dada pela equação (2.3).

A partir da teoria de curvas (ver [7]) é possível escrever tanto a curva de Bézier de ordem 3 quanto o seu vetor tangente pelas seguintes equações, respectivamente:

$$\mathbf{C}_k(t) = \sum_{i=0}^3 \mathbf{V}_{k,i} B_{i,3}(t), \quad 0 \leq t \leq 1, \quad k = 0, 1, 2, \quad (2.4)$$

$$\frac{d}{dt} \mathbf{C}_k(t) = 3 \sum_{i=0}^2 \mathbf{W}_{k,i} B_{i,2}(t), \quad 0 \leq t \leq 1, \quad k = 0, 1, 2, \quad (2.5)$$

com $V_{k,0} = Q_k$, $V_{k,3} = Q_{k+1}$ (por definição, $Q_3 := Q_0$) e os demais vértices de controle $V_{k,i}$ ($i = 1, 2$) determinados mais adiante pelo teorema 1. Quanto ao termo $W_{k,i}$ da equação (2.5) é dado por $W_{k,i} = \Delta V_{k,i} = V_{k,i+1} - V_{k,i}$ $i = 0, 1$ e 2 . Tanto os vértices de controle $V_{k,i}$ quanto os vetores $W_{k,i}$ são ilustrados na figura 1.

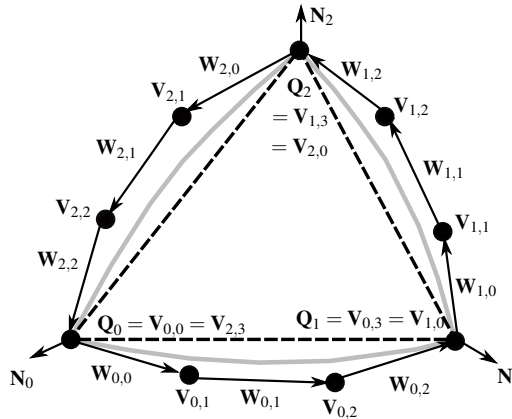


Figura 1: Curvas de contorno de ordem 3.

A utilização de curvas de ordem 3 é motivada pelo fato de que esta ordem é a menor a ser considerada para a construção de curvas polinomiais não coplanares [21]. O recurso das superfícies quárticas, a partir das curvas de contorno cúbica, é justificado pelo fato desta ordem ser a menor para a construção de superfícies triangulares de Bézier (ou de Gregory) com continuidade G^1 [17]. Assim, de acordo com a equação (2.3), pode ser escrita a superfícies de interesse, em coordenadas baricêntricas, da seguinte forma:

$$S(u, v, w) = \sum_{i+j+k=4} p_{i,j,k} \frac{4!}{i!j!k!} u^i v^j w^k, \quad u, v, w \geq 0; \quad u + v + w = 1; \quad i, j, k \geq 0 \tag{2.6}$$

com

$$p_{0,0,4} = Q_0, \quad p_{0,4,0} = Q_1, \quad p_{4,0,0} = Q_2.$$

Ao contrário das curvas de [6] ou das curvas de quarta ordem proposta por [10], as curvas utilizadas neste trabalho, assim como as desenvolvidas em [21], não são restritas às planares. A estratégia, utilizada neste trabalho para a construção dos elementos triangulares G^1 , é dependente do ajuste das curvas cúbicas de Bézier entre os pares de pontos amostrados ($Q_0 - Q_1$, $Q_1 - Q_2$, $Q_2 - Q_0$) sendo que, em seguida o grau dessas curvas é elevado para ordem quatro, de modo que seja usada para desenvolver a superfície composta por elementos de quarta ordem de Gregory. As relações entre os vértices de controle de Bézier com os pontos normais interpolantes são dadas pelo teorema que segue. Este teorema foi provado em [22] e é utilizado para determinar os termos $V_{k,i}$, $i = 1$ e 2 , presentes na equação (2.4).

Teorema 2.1. *Seja a curva de contorno cúbica dada por $C_k(t)$. Considere, ainda, por definição os seguintes parâmetros $\mathbf{d}_k = \|\mathbf{V}_{k,3} - \mathbf{V}_{k,0}\|$, $\Gamma_k = \|\mathbf{V}_{k,3} - \mathbf{V}_{k,0}\|/\mathbf{d}_i$, $a_k = \mathbf{N}_k \cdot \mathbf{N}_{k+1}$, $a_{k,0} = \mathbf{N}_k \cdot \Gamma_k$ e $a_{k,1} = \mathbf{N}_{k+1} \cdot \Gamma_k$. Assuma que se $a_{k,0} = 0$ e $a_{k,1} \neq 0$ (ou se $a_{k,0} \neq 0$ e $a_{k,1} = 0$) então $a_k > 0$. Desta forma, a curva cúbica de Bézier (equação 2.4) fica descrita por*

$$\begin{aligned} \mathbf{V}_{k,0} &= \mathbf{Q}_k, \\ \mathbf{V}_{k,3} &= \mathbf{Q}_{k+1}, \\ \mathbf{V}_{k,1} &= \mathbf{V}_{k,0} + \mathbf{d}_k (6\Gamma_k - 2\zeta_k \mathbf{N}_k + \vartheta_k \mathbf{N}_{k+1})/18, \\ \mathbf{V}_{k,2} &= \mathbf{V}_{k,3} - \mathbf{d}_k (6\Gamma_k + \zeta_k \mathbf{N}_k - 2\vartheta_k \mathbf{N}_{k+1})/18, \end{aligned}$$

com

$$\begin{aligned} \zeta_k &= 6(2a_{k,0} + a_k a_{k,1}) / (4 - a_k^2), \\ \vartheta_k &= 6(2a_{k,1} + a_k a_{k,0}) / (4 - a_k^2). \end{aligned}$$

Os termos $\mathbf{V}_{k,i}$ ($i = 0, \dots, 3$) estão entre \mathbf{Q}_k e \mathbf{Q}_{k+1} (ver figura 1) e as direções de suas normais principais são paralelas a \mathbf{N}_k em $t = 0$ e paralela a \mathbf{N}_{k+1} em $t = 1$.

Como mencionado anteriormente, quaisquer dois ou mais elementos adjacentes são unidos com continuidade G^1 se tiverem tanto um plano tangente quanto uma curva de contorno em comum. Uma abordagem utilizada por diversos autores [7, 17, 18, 5, 11] para a composição de elementos de superfícies G^1 é idealizar faixas tangentes ao longo de cada curva de contorno e então construir o elemento de superfície de tal forma que as derivadas que cruzam as curvas de contorno estejam nas direções $(1, -1/2, -1/2)$, $(-1/2, -1/2, 1)$ e $(-1/2, 1, -1/2)$. Em coordenadas baricêntricas, esta faixa tangente é formada pela derivada direcional fornecida pelo vetor

$$\mathbf{F}_k(t) = \sum_{i=0}^3 \mathbf{D}_{k,i} B_{i,3}(t), \quad k = 0, 1, 2, \quad 0 \leq t \leq 1, \tag{2.7}$$

com

$$\mathbf{D}_{0,i} = \mathbf{P}_{1,i,3-i} - \frac{1}{2}(\mathbf{P}_{0,i+1,3-i} - \mathbf{P}_{0,i,4-i}), \tag{2.8}$$

$$\mathbf{D}_{1,i} = \mathbf{P}_{i,3-i,1} - \frac{1}{2}(\mathbf{P}_{i+1,3-i,0} - \mathbf{P}_{i,4-i,0}), \tag{2.9}$$

$$\mathbf{D}_{2,i} = \mathbf{P}_{3-i,1,i} - \frac{1}{2}(\mathbf{P}_{3-i,0,i+1} - \mathbf{P}_{4-i,0,i}). \tag{2.10}$$

Assim como utilizado por [21], o presente trabalho também utiliza a ideia de faixas tangentes, entretanto, ao invés de utilizar os vetores tangente e binormal da estrutura de Frenet para gerar tais faixas, foram usados o vetor tangente (ver equação 2.5) e o vetor

$$\mathbf{H}_k(t) = \sum_{k=0}^2 A_{k,j} B_{j,2}(t), \quad 0 \leq t \leq 1, \quad k = 0, 1, 2 \tag{2.11}$$

em que

$$\begin{aligned} A_{k,0} &= N_k \times \frac{W_{k,0}}{\|W_{k,0}\|}, \\ A_{k,2} &= N_{k+1} \times \frac{W_{k,2}}{\|W_{k,2}\|}, \\ A_{k,1} &= \frac{A_{k,0} + A_{k,2}}{\|A_{k,0} + A_{k,2}\|}. \end{aligned} \tag{2.12}$$

Na figura 2 é mostrada a construção das faixas tangentes a partir dos vetores tangente, $d\mathbf{C}(t)/dt$, e do vetor \mathbf{H}_k definido na equação (2.11).

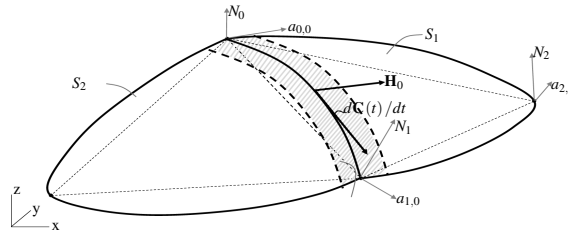


Figura 2: Faixa gerada pelo vetor tangente $d\mathbf{C}(t)/dt$ e pelo vetor \mathbf{H}_k .

A partir da derivada direcional, acima descrita, é possível construir elementos triangulares com o uso das superfícies de Gregory [5]. Os pontos de controle das curvas de contorno quártica $\mathbf{C}(t)$ (grau elevado a partir da cúbica), que são construídos a partir dos novos vértices de controle $\mathbf{L}_{k,j}$ e obtidos pela expressão $\mathbf{L}_{k,j} = 1/4 [j\mathbf{V}_{k,j-1} + (4-j)\mathbf{V}_{k,j}]$, são usados como pontos das bordas do elemento triangular. Para o elemento de Bézier, os pontos de controle internos adjacentes à borda, por exemplo $\mathbf{P}_{1,1,2}$ e $\mathbf{P}_{1,2,1}$ com respeito ao contorno entre \mathbf{Q}_0 e \mathbf{Q}_1 , podem ser obtidos a partir da imposição da restrição de continuidade tangente através de suas bordas. Isto implica que cada ponto de controle interno é determinado duas vezes, geralmente com posições distintas, uma vez para cada contorno com o qual está associado (ver figura 3). Chiyokura [5] resolve esta situação, conforme Gregory [8], com o uso de uma combinação entre as duas posições de tal forma que assegure continuidade do plano tangente na interface entre os elementos triangulares de Bézier. Com o uso dos pontos de controle de Gregory, é possível escrever os pontos internos de forma única e que satisfaçam a continuidade desejada por meio das relações

$$\mathbf{P}_{1,1,2} = \frac{1}{u+v} (u\mathbf{G}_{2,2} + v\mathbf{G}_{0,1}), \tag{2.13}$$

$$\mathbf{P}_{1,2,1} = \frac{1}{w+u} (w\mathbf{G}_{0,2} + u\mathbf{G}_{1,1}), \tag{2.14}$$

$$\mathbf{P}_{2,1,1} = \frac{1}{v+w} (v\mathbf{G}_{1,2} + w\mathbf{G}_{2,1}). \tag{2.15}$$

Os pontos $\mathbf{G}_{k,1}$ e $\mathbf{G}_{k,2}$, com $k = 0, 1, 2$, mostrados na figura 3, são determinados a partir da restrição de que as derivadas direcionais, que cruzam cada contorno, pertençam à faixa tangente (ver figura 2) construída para cada contorno correspondente, isto é,

$$\mathbf{F}_k(t) = \frac{1}{3} \alpha_k(t) \frac{d}{dt} \mathbf{C}(t) + \beta_k(t) \mathbf{H}_k(t), \quad k = 0, 1, 2 \tag{2.16}$$

na qual $\alpha_k(t)$ e $\beta_k(t)$ são polinômios em t , $\mathbf{F}_k(t)$ é definido pela equação (2.7) com os valores dos $\mathbf{P}_{i,j,k}$ das equações (2.8-2.10) substituídos pelos valores apropriados das equações (2.13-2.15) e os vetores $\frac{d}{dt} \mathbf{C}(t)$ e \mathbf{H}_k são dados pelas equações (2.5) e (2.11), respectivamente.

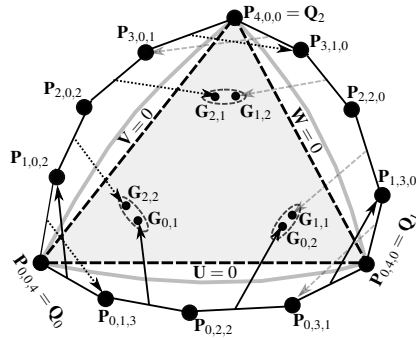


Figura 3: Pontos de controle de Gregory.

Uma vez que $F_k(t)$ é cúbica (ver equação 2.7) e tanto $\frac{d}{dt}C_k(t)$ quanto $H_k(t)$ são quadráticas (ver equações 2.5 e 2.11, respectivamente), é necessário compatibilizar a ordem das funções escritas na equação (2.16) adotando ordem um (linear) para $\alpha_k(t)$ e $\beta_k(t)$ da seguinte forma

$$\begin{aligned} \alpha_k(t) &= \lambda_{k,0}B_{0,1}(t) + \lambda_{k,1}B_{1,1}(t), \\ \beta_k(t) &= \mu_{k,0}B_{0,1}(t) + \mu_{k,1}B_{1,1}(t), \end{aligned} \tag{2.17}$$

com $0 \leq t \leq 1, i = 0, 1, 2$.

Decorre das equações (2.5, 2.7, 2.11 e 2.17) que a equação (2.16) pode ser escrita como

$$\begin{aligned} \sum_{j=0}^3 \mathbf{D}_{k,j}B_{j,3}(t) &= \sum_{j=0}^1 \lambda_{1,j}B_{j,3}(t) \sum_{j=0}^2 \mathbf{W}_{k,j}B_{j,2}(t) + \\ &\sum_{j=0}^1 \mu_{k,j}B_{j,1}(t) \sum_{j=0}^2 \mathbf{A}_{k,j}B_{j,2}(t) \end{aligned} \tag{2.18}$$

ou, com o uso da relação [12]

$$B_{i,m}(t)B_{j,n}(t) = \frac{\binom{m}{i} \binom{n}{j}}{\binom{m+n}{i+j}} B_{i+j,m+n}(t) \tag{2.19}$$

a equação (2.18) pode ser reescrita

$$\sum_{j=0}^3 \mathbf{D}_{k,j}B_{j,3}(t) = \sum_{j=0}^1 \sum_{i=0}^2 \frac{\binom{1}{j} \binom{2}{i}}{\binom{3}{j+i}} (\lambda_{k,j} \mathbf{W}_{k,i} + \mu_{k,j} \mathbf{A}_{k,i}) B_{j+i,3}(t) \tag{2.20}$$

com a notação $(u, v) := \frac{u!}{v!(u-v)!}$.

Comparando os coeficientes de $B_{j,3}(t)$, $j = 0, \dots, 3$, na Eq.(2.20), tem-se

$$\begin{aligned} D_{k,0} &= \lambda_{k,0} \mathbf{W}_{k,0} + \mu_{k,0} \mathbf{A}_{k,0}, \\ D_{k,1} &= \frac{2}{3} \lambda_{k,0} \mathbf{W}_{k,1} + \frac{1}{3} \lambda_{k,1} \mathbf{W}_{k,0} + \frac{2}{3} \mu_{k,0} \mathbf{A}_{k,1} + \frac{1}{3} \mu_{k,1} \mathbf{A}_{k,0}, \\ D_{k,2} &= \frac{1}{3} \lambda_{k,0} \mathbf{W}_{k,2} + \frac{2}{3} \lambda_{k,1} \mathbf{W}_{k,1} + \frac{1}{3} \mu_{k,0} \mathbf{A}_{k,2} + \frac{2}{3} \mu_{k,1} \mathbf{A}_{k,1}, \\ D_{k,3} &= \lambda_{k,1} \mathbf{W}_{k,2} + \mu_{k,1} \mathbf{W}_{k,2}. \end{aligned} \quad (2.21)$$

Os vetores $\mathbf{D}_{k,0}$ e $\mathbf{D}_{k,3}$, $k = 0, 1, 2$ são conhecidos e pertencem aos planos normais a \mathbf{N}_k e \mathbf{N}_{k+1} , respectivamente. A determinação de $\lambda_{k,0}$, $\mu_{k,0}$, $\lambda_{k,1}$ e $\mu_{k,1}$ é feita de forma imediata a partir da primeira e última expressão da equação (2.21). Já os vetores $G_{k,1}$ e $G_{k,2}$, $i = 0, 1, 2$, são obtidos a partir da segunda e terceira expressão da equação (2.21), além da derivadas nas direções cruzadas entre as bordas dos elementos triangulares presentes nas equações (2.8-2.10).

3 PROCEDIMENTO PARA CONSTRUÇÃO DE ELEMENTOS TRIANGULARES G^1

Após a apresentação teórica para construção de superfícies suaves G^1 , nesta seção é detalhado o procedimento prático da construção dos elementos triangulares de grau quatro com continuidade geométrica. De forma resumida, é apresentado na figura 4 o fluxograma geral para construção dos elementos triangulares. Nos itens abaixo é detalhada cada etapa.

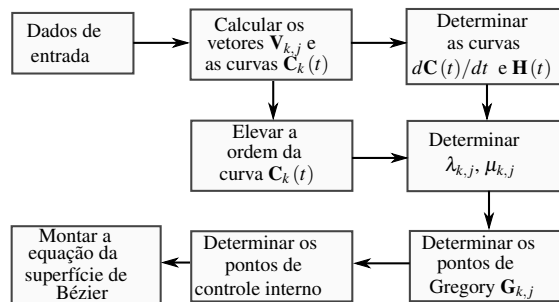


Figura 4: Fluxograma para a criação de elementos triangulares G^1 .

- **Etapa 1: Dados de entrada;** Nesta etapa são fornecidas as coordenadas geométricas (\mathbf{Q}_0 , \mathbf{Q}_1 , \mathbf{Q}_2) e as normais (\mathbf{N}_0 , \mathbf{N}_1 , \mathbf{N}_2) nos vértices do triângulo como mostrado na figura 1;
- **Etapa 2: Calcular os vetores $\mathbf{V}_{k,j}$ e as curvas $C_k(t)$;** Nesta etapa é utilizado o teorema 2.1 para ajustar as curvas de Bézier aos pares de pontos $(\mathbf{Q}_0, \mathbf{Q}_1)$, $(\mathbf{Q}_1, \mathbf{Q}_2)$ e $(\mathbf{Q}_2, \mathbf{Q}_0)$, além de determinar os vetores $\mathbf{V}_{k,j}$ e $C_k(t)$, com $k, j = 0, 1, 2$ conforme o algoritmo apresentado na figura 5.
- **Etapa 3: Determinar as curvas $\frac{d}{dt}C(t)$ e $\mathbf{H}(t)$;** A partir da equação (2.5) é calculado $\frac{d}{dt}C(t)$ e das equações (2.11) e (2.12) é determinada a curva $\mathbf{H}(t)$. O algoritmo para o cálculo desta etapa é apresentado na figura 6.

Dados de entrada:
 $\mathbf{N}_k, \mathbf{Q}_k$, com $k = 0, 1, 2$
 Início
 Passo 1: Leia $\mathbf{N}_k, \mathbf{Q}_k$ para $k = 0, 1, 2$
 Para $k \leftarrow 0$ a 2 faça
 Passo 2: $\mathbf{V}_{k,0} \leftarrow \mathbf{Q}_k$ e $\mathbf{V}_{k,3} \leftarrow \mathbf{Q}_{k+1}$ com $\mathbf{Q}_3 = \mathbf{Q}_0$
 Passo 3: $\mathbf{d}_k \leftarrow \|\mathbf{V}_{k,3} - \mathbf{V}_{k,0}\|$
 Passo 4: $\Gamma_k \leftarrow \|\mathbf{V}_{k,3} - \mathbf{V}_{k,0}\|/\mathbf{d}_k$
 Passo 5: $a_{k,0} \leftarrow \mathbf{N}_k \cdot \Gamma_k$
 Passo 6: $a_k \leftarrow \mathbf{N}_k \cdot \mathbf{N}_{k+1}$
 Passo 7: $a_{k,1} \leftarrow \mathbf{N}_{k+1} \cdot \Gamma_k$
 Passo 8: $\zeta_k \leftarrow 6(2a_{k,0} + a_k a_{k,1}) / (4 - a_k^2)$
 $\vartheta_k \leftarrow 6(2a_{k,1} + a_k a_{k,0}) / (4 - a_k^2)$
 Passo 9: $\mathbf{V}_{k,1} \leftarrow \mathbf{V}_{k,0} + \mathbf{d}_k(6\Gamma_k - 2\zeta_k \mathbf{N}_k + \vartheta_k \mathbf{N}_{k+1})/18$
 Passo 10: $\mathbf{V}_{k,2} \leftarrow \mathbf{V}_{k,3} - \mathbf{d}_k(6\Gamma_k + \zeta_k \mathbf{N}_k - 2\vartheta_k \mathbf{N}_{k+1})/18$
 Passo 11: $\mathbf{C}_k(t) \leftarrow \mathbf{V}_{k,i} \mathbf{B}_{i,3}(t), 0 \leq t \leq 1, i = 0, 1, 2$
 Fim para
 Fim

Figura 5: Algoritmo de cálculo dos vetores $\mathbf{V}_{k,j}$ e das curvas $\mathbf{C}_k(t)$.

Dados de entrada:
 Passo 1, Passo 9, Passo 10
 Início
 Para $k \leftarrow 0$ a 2 faça
 Passo 12: $\mathbf{A}_{k,0} \leftarrow \mathbf{N}_k \times \frac{\mathbf{W}_{k,0}}{\|\mathbf{w}_{k,0}\|}$
 $\mathbf{A}_{k,2} \leftarrow \mathbf{N}_{k+1} \times \frac{\mathbf{W}_{k,2}}{\|\mathbf{w}_{k,2}\|}$
 Passo 13: $\mathbf{A}_{k,1} \leftarrow \frac{\mathbf{A}_{k,0} + \mathbf{A}_{k,2}}{\|\mathbf{A}_{k,0} + \mathbf{A}_{k,2}\|}$
 Passo 14: $\mathbf{H}_k(t) \leftarrow \sum_{j=0}^2 \mathbf{A}_{k,j} \mathbf{B}_{j,2}(t)$
 Passo 15: $\mathbf{W}_{k,i} \leftarrow \Delta \mathbf{V}_{k,i}$, com $\Delta \mathbf{V}_{k,i} \leftarrow \mathbf{V}_{k,i+1} - \mathbf{V}_{k,i}$ e $i \leftarrow 0, 1, 2$
 Passo 16: $\frac{d\mathbf{C}_k(t)}{dt} \leftarrow 3 \sum_{i=0}^2 \mathbf{W}_{k,i} \mathbf{B}_{i,2}(t)$
 Fim Fim para

Figura 6: Algoritmo para determinar as curvas $\frac{d}{dt} \mathbf{C}(t)$ e $\mathbf{H}(t)$.

- **Etapa 4: Elevar a ordem da curva $\mathbf{C}_k(t)$** ; O procedimento para elevar a ordem de 3 para 4 o grau da curva de contorno é mostrado na figura 7.
- **Etapa 5: Determinar os parâmetros $\lambda_{k,j}$ e $\mu_{k,j}$, com $k = 0, 1, 2; j = 0, 1$** ; O procedimento para o cálculo dos parâmetros $\lambda_{k,j}$ e $\mu_{k,j}$ a partir da equação (2.21) é explicitamente apresentado na figura 8.
- **Etapa 6: Determinar os pontos de Gregory $\mathbf{G}_{k,j}$, de controle interno e superfície de Bézier**; O procedimento, apresentado na figura 9, determina os pontos de Gregory (ver equações 2.8 - 2.10 e 2.21), de controle interno $\mathbf{P}_{112}, \mathbf{P}_{121}, \mathbf{P}_{211}$ (ver equações 2.13 - 2.15), e por fim, é feita a construção do elemento triangular de continuidade G^1 (ver equação 2.6).

Dados de entrada:
 Passo 1, Passo 9, Passo 10
 Início
 Para $k \leftarrow 0$ a 2 faça
 Para $j \leftarrow 0$ a 4 faça
 Passo 17: $\mathbf{L}_{k,j} \leftarrow \frac{1}{4} [j\mathbf{V}_{k,j-1} + (4-j)\mathbf{V}_{k,j}]$
 Passo 18: $\mathbf{P}_{0,j,4-j} \leftarrow \mathbf{L}_{0,j}$, $\mathbf{P}_{j,4-j,0} \leftarrow \mathbf{L}_{1,j}$, $\mathbf{P}_{4-j,0,j} \leftarrow \mathbf{L}_{2,j}$
 Fim do para
 Fim do para
 Fim

Figura 7: Algoritmo para elevação da ordem da curva $\mathbf{C}_k(t)$.

Dados de entrada:
 Passo 12, Passo 15, Passo 18
 Início
 Passo 19: $\mathbf{D}_{0,0} \leftarrow \mathbf{P}_{1,0,3} - \frac{1}{2}(\mathbf{P}_{0,1,3} + \mathbf{P}_{0,0,4})$, $\mathbf{D}_{1,0} \leftarrow \mathbf{P}_{0,3,1} - \frac{1}{2}(\mathbf{P}_{1,3,0} + \mathbf{P}_{0,4,0})$
 $\mathbf{D}_{2,0} \leftarrow \mathbf{P}_{3,1,0} - \frac{1}{2}(\mathbf{P}_{3,0,1} + \mathbf{P}_{4,0,0})$, $\mathbf{D}_{0,3} \leftarrow \mathbf{P}_{1,3,0} - \frac{1}{2}(\mathbf{P}_{0,4,0} + \mathbf{P}_{0,3,1})$
 $\mathbf{D}_{1,3} \leftarrow \mathbf{P}_{3,0,1} - \frac{1}{2}(\mathbf{P}_{4,0,0} + \mathbf{P}_{3,1,0})$, $\mathbf{D}_{2,3} \leftarrow \mathbf{P}_{0,1,3} - \frac{1}{2}(\mathbf{P}_{0,0,4} + \mathbf{P}_{1,0,3})$
 Para $k \leftarrow 0$ a 2 faça
 Passo 20: $\mu_{k,0} \leftarrow \mathbf{D}_{k,0} \cdot \mathbf{A}_{k,0}$, $\mu_{k,1} \leftarrow \mathbf{D}_{k,3} \cdot \mathbf{A}_{k,2}$
 Passo 21: $\lambda_{k,0} \leftarrow \frac{\mathbf{D}_{k,0} \cdot \mathbf{W}_{k,0}}{\mathbf{W}_{k,0} \cdot \mathbf{W}_{k,0}}$, $\lambda_{k,1} \leftarrow \frac{\mathbf{D}_{k,3} \cdot \mathbf{W}_{k,2}}{\mathbf{W}_{k,2} \cdot \mathbf{W}_{k,2}}$
 Fim para
 Fim

Figura 8: Algoritmo para cálculo dos parâmetros $\lambda_{k,j}$ e $\mu_{k,j}$.

4 AUXÍLIO DO SOFTWARE BLENDERTM NA CONSTRUÇÃO DE ELEMENTOS G^1

Nas figuras 4 - 9 foram apresentados os procedimentos para a criação de superfícies suaves propostos neste trabalho. Dentre as diversas etapas, há uma de primordial importância para a construção eficiente das geometrias com continuidade do plano tangente, que é a etapa 1: entrada de dados. Nesta etapa são exigidas as coordenadas geométricas e os vetores normais nos vértices da malha triangular plana (figura 1), que podem ser de difícil obtenção para geometrias complexas. Para contornar esta dificuldade, o presente trabalho acoplou o software de computação gráfica denominado *BlenderTM* ao programa, desenvolvido na linguagem Fortran, para construção das superfícies G^1 .

O *BlenderTM* é um software gráfico de código aberto, escrito nas linguagens C, C++ e Python, desenvolvido para animação e criação de filmes, além de desenhos animados, sem nenhum objetivo, a princípio, para a análise numérica. A partir da criação de geometrias das mais variadas complexidades no *BlenderTM*, é possível extrair deste software as informações de entrada e então seguir todas as etapas descritas nas figuras 4 - 9. Cabe salientar, que as coordenadas e os vetores normais são obtidos do programa *BlenderTM* a partir de malhas triangulares de três nós, o que torna a sua aplicação direta para análise numérica limitada às aproximações de grau 1 e sem garantia de continuidade do plano tangente na zona de interface entre os elementos da malha. Salientando ainda que, mesmo utilizando malha triangular de três nós, a normal obtida é única, uma vez que esta é calculada a partir do processo de parametrização local (em cada trecho de

Dados de entrada:
 Passo 12, Passo 13, Passo 15, Passo 17, Passo 18, Passo 20, Passo 21

Início

Para $k \leftarrow 0$ a 2 faça

Passo 22: $\mathbf{G}_{k,1} \leftarrow \frac{1}{2}(\mathbf{L}_{k,1} + \mathbf{L}_{k,2}) + \frac{2}{3}\lambda_{k,0}\mathbf{W}_{k,1} + \frac{1}{3}\lambda_{k,1}\mathbf{W}_{k,0} + \frac{2}{3}\mu_{k,0}\mathbf{A}_{k,1} + \frac{1}{3}\mu_{k,1}\mathbf{A}_{k,0}$
 $\mathbf{G}_{k,2} \leftarrow \frac{1}{2}(\mathbf{L}_{k,2} + \mathbf{L}_{k,3}) + \frac{1}{3}\lambda_{k,0}\mathbf{W}_{k,2} + \frac{2}{3}\lambda_{k,1}\mathbf{W}_{k,1} + \frac{1}{3}\mu_{k,1}\mathbf{A}_{k,2} + \frac{2}{3}\mu_{k,1}\mathbf{A}_{k,1}$

Fim Para

Passo 23: $\mathbf{P}_{1,2,1} \leftarrow \frac{1}{w+u}(w\mathbf{G}_{0,2} + u\mathbf{G}_{1,1})$
 Passo 24: $\mathbf{P}_{1,1,2} \leftarrow \frac{1}{u+v}(u\mathbf{G}_{2,2} + v\mathbf{G}_{0,1})$
 Passo 25: $\mathbf{P}_{2,1,1} \leftarrow \frac{1}{v+w}(v\mathbf{G}_{1,2} + w\mathbf{G}_{2,1})$
 Passo 26: $\mathbf{S}(u, v, w) \leftarrow \sum_{i+j+k=4} \mathbf{P}_{i,j,k} \frac{4!}{i!j!k!} u^i v^j w^k$

Fim

Figura 9: Algoritmo para determinar os pontos de Gregory, os pontos de controle interno e a superfície de Bézier.

superfícies – patch) utilizando superfície NURBS (Non Uniform Rational Basis Spline) interna do software.

Diante das limitações proporcionadas pelo *BlenderTm* para a análise numérica, a presente formulação de construção de elementos G^1 torna ideal ao acoplamento com tal software. A figura 10 apresenta o código desenvolvido em Python para extrair as informações necessárias para a etapa 1 da presente formulação de construção dos elementos G^1 .

```
import bpy
obj = bpy.data.objects["Nome_do_objeto "]

file = open("C:/Users/nome_do_arquivo", "w")
file.write("ind;x;y;z;nx;ny;nz\n")
for vert in obj.data.vertices:
    texto=str(vert.index)+' '+str(vert.co.to_tuple())[1:-1]+\
    ' '+str(vert.normal.to_tuple())[1:-1]+' \ n'
    texto=texto.replace(' ',';')
    texto=texto.replace(';','\n')
    file.write(texto)
file.write("\nind;nd0;nd1;nd2\n")
for poly in obj.data.polygons:
    texto=str(poly.index)
    for vert in poly.vertices:
        texto+=str(vert)
    texto+='\n'
    file.write(texto)

file.close()
```

Figura 10: Código fonte, em Python, para a extração das coordenadas geométricas e dos vetores normais no *BlenderTm*.

5 EXEMPLOS DE APLICAÇÃO

Para a verificação do procedimento de criação das superfícies, descrito acima, foram adotados os critérios de continuidade do plano tangente e da representatividade geométrica.

5.1 Descrição geométrica

A seguir são apresentadas as geometrias globalmente paramétricas e não-paramétricas (ou localmente paramétrica) analisadas. As características geométricas para as superfícies paramétricas toroidal e esférica discretizadas, respectivamente, com 200 e 320 elementos triangulares são apresentadas na figura 11.

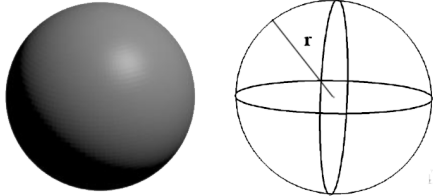
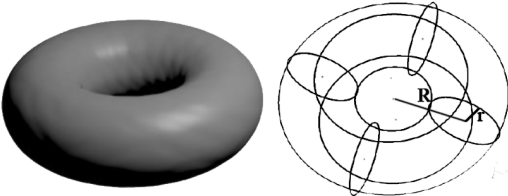
	Geometria paramétrica	Parâmetro	Aproximação
Esfera		$r=1,0$	Phong
			Nagata
Toróide		$R=1,0$ $r=0,5$	NLSA
			PN
			Presente
			Trabalho
			Trabalho

Figura 11: Características geométrica e tipo de aproximação para a construção de geometrias paramétricas auxiliada pelo *BlenderTM*.

Quando se deseja representar formas presentes na natureza, as geometrias paramétricas tornam-se bastante limitadas e são necessárias a utilização de geometrias globalmente não-paramétricas complexas. Como aplicação de geometria globalmente não-paramétrica, este trabalho analisou a continuidade do plano tangente para a geometria de uma cabeça humana (ver figura 12), a qual foi discretizada com 8704 elementos triangulares. Esta geometria possui alta complexidade, principalmente, por ter curvas acentuadas e suaves, o que a torna um grande desafio na integração CAD (Computer-Aided Designer) e CAE (Computer-Aided Engineering), principalmente pelo Método dos Elementos de Contorno.

5.2 Continuidade do plano tangente

Nesta seção é analisada a capacidade que as funções aproximadoras (Bézier G^1 , Phong, Nagata, NLSA e PN) possuem em representar continuidade dos planos tangentes nas interfaces dos elementos triangulares, ou seja, a capacidade que cada função aproximadora possui em gerar normal

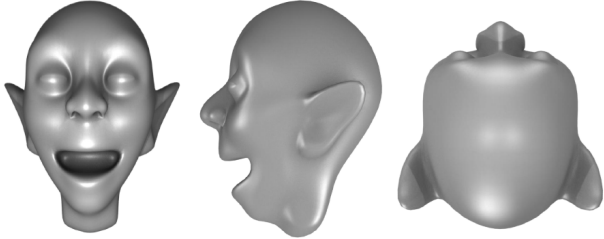
	Geometria não paramétrica	Elementos / Nós
Cabeça		8704 / 4352

Figura 12: Características geométrica, número de elementos e de nós para a geometria complexa globalmente não-paramétrica construída com auxílio do *BlenderTM*.

única na interface entre os elementos. Para realizar esta análise, é calculado o vetor normal pela equação

$$\mathbf{N}(s, t) = \frac{\mathbf{S}_s(s, t) \times \mathbf{S}_t(s, t)}{\|\mathbf{S}_s(s, t) \times \mathbf{S}_t(s, t)\|} \tag{5.1}$$

em que $\mathbf{S}_s(s, t)$ e $\mathbf{S}_t(s, t)$ são, respectivamente, vetores tangentes nas direções das curvas s - e t -paramétricas, aplicados nos pontos localizados na fronteira entre os elementos.

Dado um ponto na interface entre dois elementos triangulares genéricos (\mathbf{S}_1 e \mathbf{S}_2), os vetores normais a estes elementos, representado, respectivamente, por \mathbf{N}_1 e \mathbf{N}_2 pertencem a um mesmo ponto na interface destas duas superfícies. Uma vez determinados os vetores normais, via *BlenderTM* é calculado o cosseno do ângulo formado pelos vetores \mathbf{N}_1 e \mathbf{N}_2 por meio da equação (5.2). Este procedimento é realizado para diversos pontos nas interfaces dos elementos pertencentes à malha, o que proporciona, assim, diversos valores para o cosseno do ângulo formado pelas normais das superfícies adjacentes.

Desta forma, torna-se necessário o uso de parâmetros estatísticos, tais como, o valor médio (equação 5.3) e o desvio padrão (equação 5.4) para analisar o desempenho das funções aproximadoras em representar superfícies suaves.

$$\cos \theta = \mathbf{N}_1 \cdot \mathbf{N}_2, \tag{5.2}$$

$$média = \frac{\sum_{i=1}^n \cos \theta_i}{n}, \tag{5.3}$$

$$DP = \frac{\sqrt{\sum_{i=1}^n (\cos \theta_i - média)^2}}{n - 1}. \tag{5.4}$$

Para ilustrar a precisão da função de Bézier G^1 em representar geometrias suaves, na figura 13 é apresentado o comportamento das componentes do vetor normal (n_x, n_y e n_z) para as superfícies paramétricas, tais como, esfera, figura 13 (a-c), e toroide, figura 13 (d-f).

A partir da figura 13, e em detalhes na figura 14, pode ser observado que as imagens apresentam uma única tonalidade de cor em ambos os lados das linhas de interface dos elementos

triangulares, ou seja, é mostrado, qualitativamente, a unicidade do vetor normal na borda do elemento.

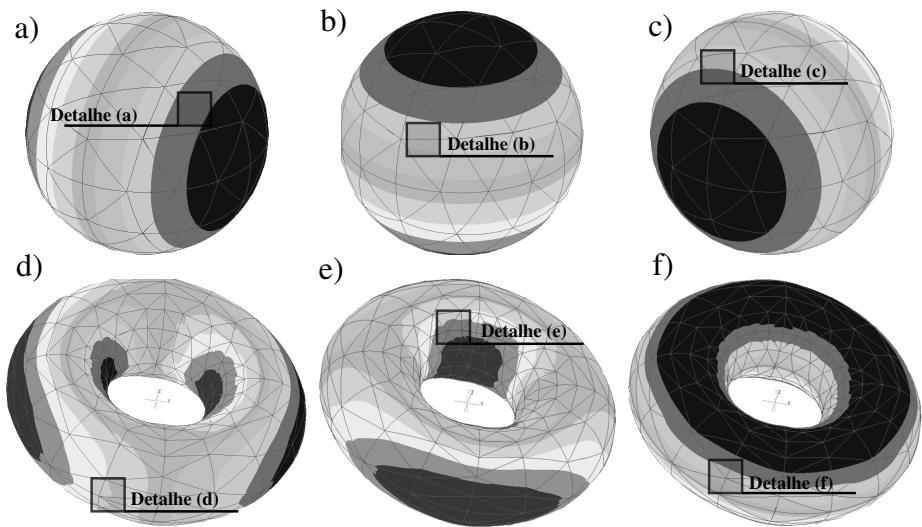


Figura 13: Componentes n_x , n_y e n_z obtidas pela aproximação de Bézier G^1 para geometria esférica (a-c) e toroidal (d-f).

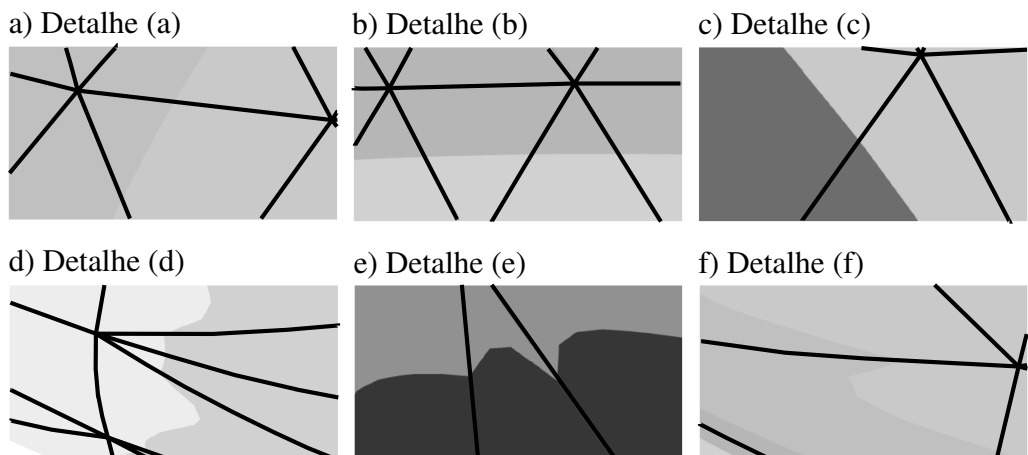


Figura 14: Detalhe das componentes do vetor normal para a esfera (a-c) e toroide (d-f). Componente n_x , (a) e (d), componente n_y , (b) e (e) e componente n_z , (c) e (f).

Outras funções aproximadoras tais como, Nagata [13], NLSA [1], PN [19] e Phong tessellation [4] são bastante difundidas na literatura [3] e possuem capacidade semelhantes às apresentadas na presente formulação quanto à construção de geometrias. No entanto, a capacidade de construção de superfícies suaves é analisada e comparada com a formulação presente (Bézier G^1).

As discretizações e os parâmetros estatísticos para as aproximações presentes na literatura [3] e no presente trabalho (chamadas de Bézier G^1) são apresentadas na tabela 1. Diante das informações presentes na tabela 1, pode-se observar a superioridade, significativa, da função Bézier G^1 frente às demais outras funções aproximadoras. Dentre estas funções, verifica-se que os triângulos de Nagata e PN possuem representatividade da continuidade do plano tangente superior aos triângulos Phong e NLSA. Devido à aplicação crescente dos triângulos de Nagata para a análise numérica ([16], [15], [9]), é chamada a atenção para a superioridade da presente proposta (Bézier G^1) frente a esta aproximação [13] quando se refere à eficiência na construção de superfícies suaves rígidas, sem consideração da deformação geométrica.

Tabela 1: Parâmetros estatísticos para as aproximações Phong, Nagata, NLSA, PN e Bézier G^1 (presente trabalho).

Número de Elementos	Ângulo entre normais de elementos adjacentes				
	Aproximação	Mínimo	Máximo	Médio	Desvio Padrão
Esfera 320	Phong [19]	0,995723	0,998740	0,997072	$7,657330(10^{-4})$
	Nagata $\xi = 0$ [19]	0,999988	1,000000	0,999996	$3,276020(10^{-6})$
	NLSA [19]	0,999958	1,000000	0,999979	$9,690380(10^{-6})$
	PN [19]	0,999975	1,000000	0,999991	$6,656300(10^{-6})$
	Presente trabalho	0,999999	1,000000	1,000000	$1,927585(10^{-16})$
Toroide 200	Phong [19]	0,915295	0,999999	0,987918	$1,578150(10^{-2})$
	Nagata $\xi = 0$ [19]	0,938828	1,000000	0,997556	$8,603600(10^{-3})$
	NLSA [19]	0,969450	1,000000	0,994738	$6,790900(10^{-3})$
	PN [19]	0,989164	1,000000	0,998873	$1,846800(10^{-3})$
	Presente trabalho	0,999999	1,000000	0,999999	$1,710914(10^{-10})$

Nas figuras 15 e 16 são apresentadas, em vista frontal e lateral, respectivamente, as variações das componentes normais (n_x, n_y, n_z) para a geometria de uma cabeça obtida pela aproximação Bézier G^1 , com os dados de entrada (coordenadas e normais nodais) obtido pelo acoplamento da formulação com o *BlenderTm*. A partir do detalhe ampliado (nariz) presente na figura 16, pode ser observado uma única tonalidade de cor na região (linha) de interseção entre elementos da malha, ou seja, obtêm-se uma única normal tanto calculado pelo elemento da esquerda quanto pelo o da direita.

A análise quantitativa foi realizada para uma discretização com 8704 elementos triangulares de Bézier G^1 . Nesta análise foram calculados os seguintes parâmetros estatísticos, referente ao cosseno do ângulo formado pelas normais na interface dos elementos: valor mínimo = 0,9999982713, valor máximo = 1,0000000000, valor médio = 0,9999999999 e desvio padrão = $6,425165(10^{-9})$. A partir destes valores é observada a excelente representatividade da continuidade do plano tangente pela aproximação Bézier G^1 que apresenta valor mínimo, médio e desvio

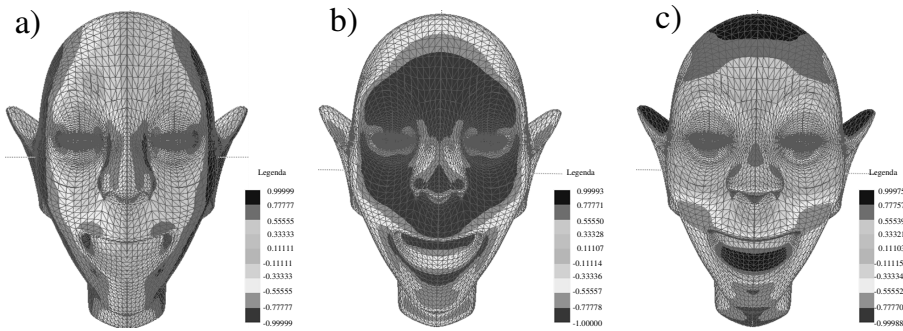


Figura 15: Geometria cabeça (vista frontal). As imagens a-c mostram, respectivamente, as componentes n_x, n_y e n_z obtidas pela aproximação de Bézier G^1 .

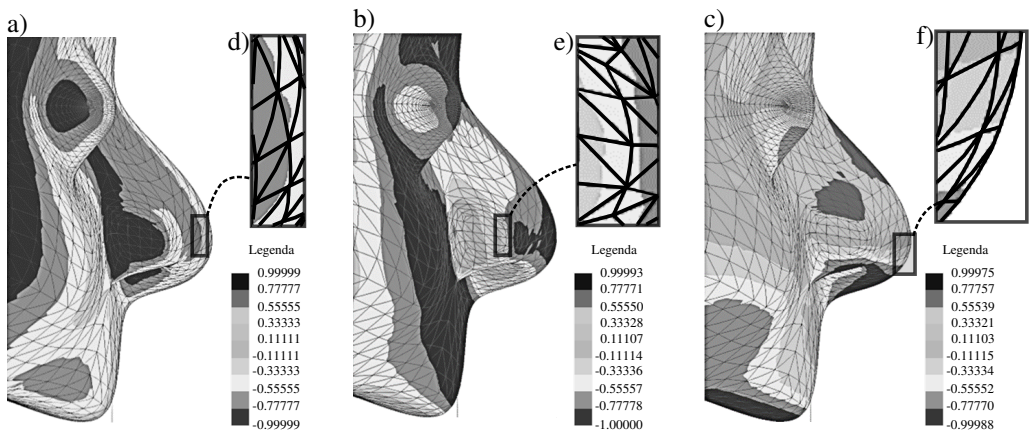


Figura 16: Geometria cabeça via aproximação Bézier G^1 (presente formulação). Componentes n_x, n_y e n_z , imagens a-c. Detalhes ampliados das componentes do vetor normal, imagens d-f.

padrão com ordem de grandeza do erro, em relação ao valor unitário, de 10^{-6} , 10^{-10} e 10^{-9} , respectivamente.

5.3 Representação geométrica

Após realizada a análise da continuidade do plano tangente na interface entre os elementos triangulares, nesta seção é verificada a capacidade em representar geometrias pela formulação, denominada, Bézier G^1 . Nesta verificação, é realizado um estudo comparativo das aproximações Phong, Nagata, NLSA e PN com a função Bézier G^1 .

Nas figuras 17 e 18 é mostrado o comportamento do parâmetro estatístico valor médio da distância entre a geometria construída pelas funções aproximadoras e a geometria analítica, à medida que é variada a quantidade de elementos. Devido a este referencial analítico, o presente trabalho realizou estudo para diversas geometrias paramétricas, entretanto, devido à sua

semelhança de resultados, são apresentadas apenas as comparações com as geometrias esférica e toroidal (figura 11).

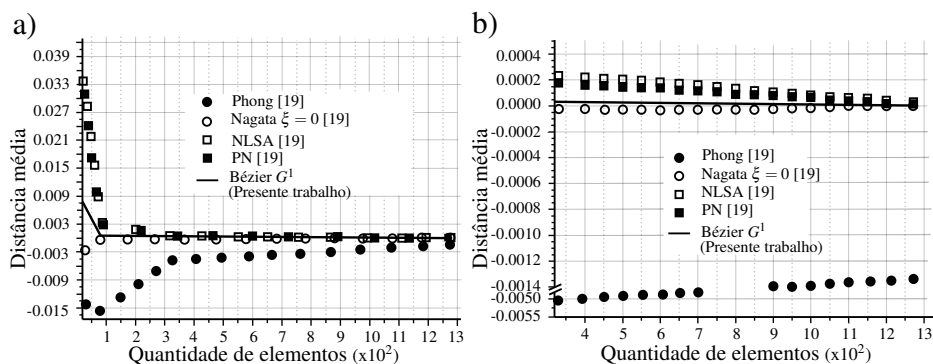


Figura 17: Valor médio da distância entre a superfície esférica analítica e a gerada pelas funções aproximadoras.

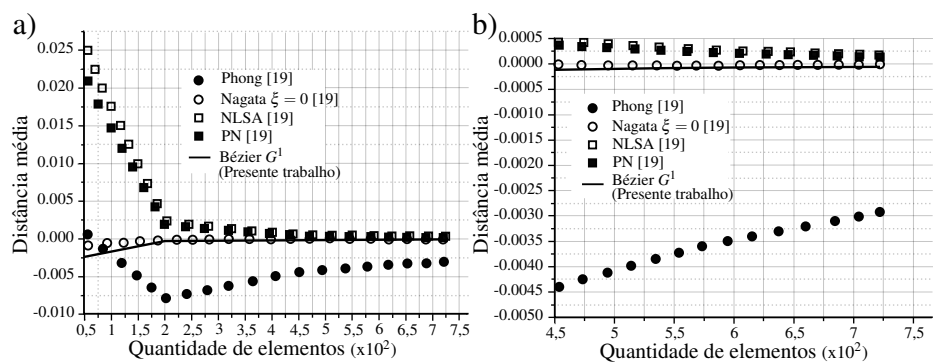


Figura 18: Valor médio da distância entre a superfície toroidal analítica e a gerada pelas funções aproximadoras.

Para a geometria esférica, figura 17 e em maior detalhe na figura 17-b, observa-se que para uma quantidade de 32 a 1300 elementos, os triângulos de Nagata e os da presente formulação (Bézier G^1) demonstram melhores resultados, enquanto a aproximação Phong apresenta o pior desempenho, com relação à distância média, quando comparado às demais aproximações na representatividade da geometria.

Na figura 18-b é apresentado, em detalhe, para uma quantidade de 50 a 720 elementos triangulares, o melhor desempenho das aproximações Bézier G^1 e de Nagata, enquanto que a função Phong, novamente, apresenta pior desempenho para o valor da distância média.

Diante dos exemplos analisados, tem-se que a função aproximadora mostrada neste trabalho (denominada de Bézier G^1), cuja principal finalidade é construir superfícies suaves, e a aproximação de Nagata apresentaram capacidade semelhante em representar geometrias.

6 CONCLUSÃO

Com o objetivo de modelar computacionalmente geometrias suaves, este trabalho apresentou uma técnica de construção de elementos triangulares com continuidade geométrica G^1 , ou seja, construção de elementos que possuem normal única nos pontos de interseção entre os elementos da malha. Esta característica única da normal é de grande importância na modelagem numérica, principalmente no uso do Método dos Elementos de Contorno, pois evita a necessidade de colocação de nós duplos para representar suavidades, o que proporciona uma maior facilidade tanto no pré quanto no pós processamento de estruturas com geométricas complexas.

Para o sucesso da técnica apresentada (Bézier G^1), são necessárias, como dado de entrada, as coordenadas geométricas e as normais dos nós, nos vértices da malha triangular. A vantagem na utilização do elemento Bézier G^1 é devido a necessidade, apenas, de criar malha formada por elementos planos, pois a função aproximadora transforma estes elementos triangulares planos em curvos. Por outro lado, é desafiador a obtenção do vetor normal para geometrias complexas. Assim, para auxiliar a obtenção desta informação foi utilizado o software de computação gráfica *BlenderTM*.

O presente trabalho acoplou, com sucesso, o software gráfico ao programa desenvolvido, em FORTRAN, para a criação de superfícies suaves G^1 , a partir do procedimento descrito acima. Como exemplo, foram apresentados resultados tanto para geometrias paramétricas (esfera e toroide) quanto para geometria globalmente não-paramétrica (cabeça). A partir da análise da continuidade do plano tangente na interface entre os elementos e da capacidade de representar formas geométricas, pôde ser observado excelente eficiência da função Bézier G^1 em representar geometrias suaves independente de sua complexidade e sem perder a eficiência na representatividade da geometria.

AGRADECIMENTOS

Agradecimentos ao CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico) e à FAPESP (Fundação de Amparo à Pesquisa do Estado de São Paulo) pelo auxílio financeiro.

ABSTRACT. In the computational modeling of solids, the construction of a smooth surface is one of the fundamental requirements in order to apply numerical methods, particularly the Boundary Element Methods. In addition, it is required in many other applications such as computer animation models and reconstruction of medical images. All these applications have the geometric continuity G^1 as the fundamental condition for generating smooth surfaces. In this study, the technique of constructing smooth geometries from boundary curves is used. For this construction, the geometric coordinates and the normals of vertices of the triangular patches are required. Because these informations are not immediately available, especially the normal vector, in this study, the computer graphics software named *BlenderTM* is used to overcome this gap in the acquisition of input data. Thus, the procedure for creating the triangular patches with continuity G^1 and the algorithm for obtaining initial information are presented. An excellent efficiency in imposing of the smoothness and ac-

curacy in geometrical representation is observed for both globally parametric and globally non-parametric surfaces.

Keywords: Geometric continuity G^1 ; triangular patch; *BlenderTM* software; complex geometry

REFERÊNCIAS

- [1] T. Barrera, A. Hast & E. Benftsson. Surface construction near least square acceleration based on vertex normal on triangular meshes. In “SIGRAD 2002” (2002), pp. 17–22.
- [2] P.E. Bézier. “Fundamental developments of computer-aided geometric modeling”. Academic Press (1993), chapter the first of CAD/CAM and the UNISURF CAD system, pp. 13–26.
- [3] M. Boshiroli, C. Fünfzig, L. Romani & G. Albrecht. A comparison of local parametric C^0 Bézier interpolants for triangular meshes. *Computers & Graphics*, **35** (2011), 20–34.
- [4] T. Boubekeur & M. Alexa. Phong tessellation. In “SIGGRAPH conference and exhibition on computer graphics and interactive techniques in Asia” (2008), pp. 1–5.
- [5] H. Chiyokura. “Advanced Computer Graphics”. Springer-Verlag (1986), chapter Localized surface interpolation method for irregular meshes, pp. 3–19.
- [6] C. Deboor, Holling & M. Sabin. High accuracy geometric Hermit interpolation. *Computer-Aided Design*, **4**(4) (1987), 269–278.
- [7] G. Farin. “Surfaces in Computer Aided Geometric Design”. North-Holand (1983), chapter Smooth interpolation to scattered 3D data, pp. 43–63.
- [8] J.A. Gregory. “Computer Aided Geometric Design”. Academic Press (1974), chapter Smooth interpolation without twist constraints, pp. 71–87.
- [9] T. Hama, T. Nagata, C. Teodosiu, A. Makinouchi & H. Takuda. Finite-element simulation of spring-back in sheet metal forming using local interpolation for tool surfaces. *International Journal of Mechanical Sciences*, **50** (2008), 175–192.
- [10] D. Hansford, R.E. Barnhill & G. Farin. Curves with quadric boundary precision. *Computer-aided Design*, **11** (1994), 519–531.
- [11] C. Loop. A G^1 triangular surface of arbitrary topological type. *Computer Aided Geometric Design*, **11** (1994), 303–330.
- [12] D. Marsh. “Applied geometry for computer graphic and CAD”. Springer (2005).
- [13] T. Nagata. Simple local interpolation of surfaces using normal vector. *Computer Aided Geometric Design*, **22**(4) (2005), 327–347.
- [14] T. Nagata. Smooth local interpolation of surfaces using normal vectors. *Journal of Applied Mathematics*, (2010).

- [15] D.M. Neto, M.C. oliveira & L.F. Menezes. “Nagata patch interpolation algorithms: strategies towards sheet metal forming tool description in CAE”. VDM Verlag (2010).
- [16] D.M. Neto, M.C. Oliveira, L.F. Menezes & J.L. Alves. Improving Nagata patch interpolation applied for tool surface description in sheet metal forming simulation. *Computer-Aided design*, **45** (2013), 639–656.
- [17] B. Piper. Visually smooth interpolation with triangular Bézier patches. In G. Farin (editor), “Geometric Modelling: Algorithms and News Trends”. SIAM (1987), pp. 221–233.
- [18] L.A. Shirman & C.H. Séquin. Local surface interpolation with Bézier patches. *Computer Aided Geometric Design*, **4**(4) (1987), 279–295.
- [19] A. Vlachos, J. Peters, C. Boyd & J.L. Mitchell. Curved PN triangles. In “Symposium on interactive 3D graphics” (2001), pp. 159–166.
- [20] D.J. Walton & D.S. Meek. Point normal interpolation for stereolithography modelling. *Computer & Graphics*, **19**(3) (1995), 345–353.
- [21] D.J. Walton & D.S. Meek. A triangular G1 patch from boundary curves. *Computer-Aided Design*, **28**(2) (1996), 113–123.
- [22] D.J. Walton & M. Yeung. Geometry modelling from CT scan for stereolithography apparatus. In Z. Tang (editor), “New advances in CAD & Computer Graphics”. International Academic Publishers (1993), pp. 417–422.