

PROPOSTA DE CÓDIGO AUTOMATIZADO PARA CONVERSÃO DE MALHAS POLIGONAIS DE FORMAS ORGÂNICAS PARA SUPERFÍCIES NURBS UTILIZÁVEIS EM SOFTWARES CAD

PROPOSAL OF AN AUTOMATED CODE FOR THE CONVERSION OF POLYGONAL MESHES OF ORGANIC SHAPES TO CAD NURBS SURFACES

Marina Pires de Castro Aguiar Vale¹, Camila Vieira Ghisleni¹, Fernanda Fontana de Gasperin¹, Roger Langone Leal¹, Fabio Pinto da Silva¹, Underléa Miotto Bruscato¹

RESUMO:

A possibilidade de digitalização 3D de elementos da natureza facilita o desenvolvimento de projetos inspirados por formas orgânicas. Entretanto, a digitalização 3D costuma produzir modelos tridimensionais constituídos por malhas poligonais, quando o padrão utilizado no projeto de produtos são as superfícies NURBS, por conta de sua precisão. Neste contexto, este trabalho propõe e analisa uma alternativa de código para conversão automatizada de malhas poligonais em superfícies NURBS para seu uso em, mas não restrito a isso, projetos de produtos baseados em formas orgânicas. Para tanto, foi criado um algoritmo de conversão por meio do *plug-in* Grasshopper[®] do *software* de modelagem Rhinoceros[®], que possibilita uma programação visual, sem a necessidade do uso de uma linguagem de programação. O código proposto possibilita a conversão automatizada de malhas poligonais de determinados formatos em superfícies NURBS, com controle de fidelidade dimensional por parte do usuário, bem como torna os modelos tridimensionais mais leves do ponto de vista computacional.

PALAVRAS-CHAVE: Grasshopper; Algoritmo; MeshToNurbs; Digitalização 3D; Modelagem paramétrica.

ABSTRACT:

The possibility of 3D scanning elements from nature facilitates the development of projects inspired by organic forms. However, 3D scanning usually produces three-dimensional models made up of polygonal meshes, while the standard used in product design is NURBS surfaces, due to their precision. In the given context, this work proposes and analyzes a code alternative for automated conversion of polygonal meshes on NURBS surfaces for use in, but not restricted to, product designs based on organic forms. For this purpose, a conversion algorithm was created using the Grasshopper[®] plug-in for modeling software Rhinoceros[®], which allows visual programming, without the need to use a programming language. The proposed code allows the automated conversion of polygonal meshes of certain formats into NURBS surfaces, with dimensional fidelity control by the user, as well as making the three-dimensional models lighter from a computational point of view.

KEYWORDS: Grasshopper; Algorithm; MeshToNurbs; 3D scanning; Parametric modeling.

¹Universidade Federal do Rio Grande do Sul

Fonte de Financiamento:
O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de financiamento 001.

Conflito de Interesse:
Declara não haver.

Ética em Pesquisa:
Declara não haver necessidade.

Submetido em: 08/02/2020
Aceito em: 30/04/2021

How to cite this article:

VALE, M. P. C. A.; GHISLENI, C. V.; GASPERIN, F. F.; LEAL, R. L.; SILVA, F. P.; BRUSCATO, U. M. Proposta de Código Automatizado para Conversão de Malhas Poligonais de Formas Orgânicas para Superfícies Nurbs Utilizáveis em Softwares CAD. *Gestão & Tecnologia de Projetos*. São Carlos, v17, n2, 2022.

<https://doi.org/10.11606/gtp.v17i2.166353>



INTRODUÇÃO

A digitalização 3D é um método que pode ser usado para muitas finalidades, como medição e visualização tridimensional, e, principalmente, para geração de modelos tridimensionais a serem retrabalhados tanto em ferramentas CAD (*Computer-Aided Design*) quanto nas BIM (*Building Information Modelling*). Uma característica da digitalização 3D é a capacidade de reproduzir, virtualmente e com fidelidade – dimensional e visual, a forma de um objeto, registrando detalhes de sua superfície, como texturas e reentrâncias. Por essa capacidade, vem sendo usada para diversos fins no projeto de produtos, desde engenharia reversa até exploração de formas. Para Palombini *et al.* (2017), o uso de tecnologias 3D, como a digitalização, pode possibilitar novas maneiras de observação e análise de elementos da natureza, bem como a fabricação e reprodução de modelos físicos desses elementos.

Entretanto, o resultado final de uma digitalização 3D costuma ser um modelo 3D constituído por uma malha poligonal (*polygonal mesh*) que, apesar de sua ampla utilização em diversas áreas da modelagem tridimensional, não é o padrão usado no desenvolvimento de produtos. Pela necessidade de precisão na produção de tais objetos, os modelos 3D neste tipo de projeto são, usualmente, constituídos por superfícies NURBS (*Non Uniform Rational Basis Spline*), que, além da precisão, na maior parte dos casos, demandam menos capacidade de processamento dos computadores.

Considerando a demanda pelo uso de formas orgânicas no processo de criação de objetos de *design*, vê-se a necessidade de automatizar o processo de transformação de malhas poligonais, obtidas a partir de digitalização 3D, em superfícies NURBS. Apesar da conversão, entre estes dois formatos, possa ser realizada, ela apresenta desafios ainda não atendidos de maneira satisfatória pelas ferramentas disponibilizadas pelos *softwares* de modelagem. Por exemplo, são detectados problemas como o aumento da carga de processamento na edição do modelo convertido. Neste contexto, um código que possibilita converter malhas poligonais pode vir a auxiliar *designers* que queiram utilizar formas orgânicas na exploração de formas em seus projetos.

Assim sendo, este trabalho tem por objetivo **propor e analisar** uma alternativa de código para conversão automatizada de malhas poligonais em superfícies NURBS para seu uso em, mas não restrito a isso, projetos de produto baseados em formas orgânicas, proporcionando ao projetista à edição do modelo tridimensional com a precisão possibilitada pelas curvas NURBS, além de reduzir o tamanho dos arquivos dos modelos tridimensionais, visando diminuir o impacto no processamento de tais modelos.

Para tanto, o experimento aqui relatado apresenta cinco etapas: (1) a seleção do objeto a ser escaneado; (2) a digitalização 3D deste objeto em um Digimil 3D (*laser scanner*); (3) a conversão do resultado da digitalização em uma malha poligonal no *software* Geomagic Qualify®; (4) a elaboração do código para conversão do modelo tridimensional obtido por meio do *software* de modelagem tridimensional Rhinoceros® e seu *plug-in* Grasshopper®, que possibilita a programação de algoritmos sem a necessidade de conhecimento de uma linguagem de programação; e (5) a análise dos resultados da conversão, considerando o desvio quadrático médio entre o modelo NURBS produzido e a malha obtida pela digitalização 3D, assim como os aspectos visuais da peça e, também, a diminuição de pontos (e conseqüente diminuição do tamanho do arquivo) entre o modelo original e o convertido.

CONSIDERAÇÕES TEÓRICAS

Por meio da digitalização 3D, faz-se possível obter amostras tridimensionais de objetos naturais para análise virtual e para prototipagem (PALOMBINI *et al.*, 2017). A biônica, ciência que busca estudar elementos e formas da natureza, bem como suas propriedades, mecanismos e estruturas (PALOMBINI *et al.* 2020; KINDLEIN JÚNIOR; GUANABARA, 2005), visando aplicação em soluções de projeto, é cada vez mais explorada no *Design*, Engenharia e Arquitetura (QUEIROZ *et al.*, 2017).

O processo de digitalização é feito, geralmente, a partir da varredura da superfície de um objeto por feixes de raios de diferentes tipos de luz. Tal processo resulta em uma nuvem de pontos que corresponde à geometria deste objeto, estabelecida pelas coordenadas cartesianas x, y e z de determinados pontos da superfície do objeto. Tal nuvem, segundo Groetelaars e Amorim (2011), é a forma mais simples de representação de uma geometria e, usualmente, precisa ser trabalhada em *softwares* específicos para geração de modelos tridimensionais. Geralmente, tais *softwares* formam polígonos a partir da união dos pontos da nuvem. A modelagem, a partir de malhas poligonais, tem planos definidos por triângulos que, por sua vez, podem ser agrupados em figuras geométricas com maior número de lados. As malhas seguem uma abordagem topológica, não sendo constituídas por uma geometria única, mas sim por vários polígonos adjacentes (TEDESCHI, 2011). As curvas, por exemplo, nas malhas poligonais, são formadas por vários segmentos de linhas retas. Dessa forma, a representação de um círculo será mais fiel ao conceito deste ente geométrico quanto maior for o número de segmentos de reta que o compuserem. Tal tipo de modelagem tridimensional, por esta característica, apesar de amplamente utilizado em diversas áreas, pode não ser o mais adequado quando se trata de projeto de produtos, pela precisão dimensional normalmente exigida nesta área.

Assim, as superfícies criadas a partir de linhas NURBS são a principal forma de modelagem tridimensional usada na produção de objetos industriais. Segundo Anderson e Crawford-Hines (2000, p. 2, tradução nossa),

Uma representação NURBS é um modelo paramétrico suave da superfície em questão que apresenta uma compactação da representação a qual oferece grande economia na manipulação, armazenamento, transmissão e uniformidade sobre uma ampla variedade de plataformas e aplicações.

Segundo Foley *et al.* (1994), as NURBS tornaram-se a forma padrão de modelagem, em programas CAD, por permitirem controle local na edição de seus pontos de controle (a edição de um ponto afeta somente a área próxima ao ponto em questão), por não variarem nas operações de transformação como, por exemplo, escalonamento e rotação, e por definirem com exatidão superfícies esféricas e elípticas.

As soluções para obtenção de superfícies NURBS, a partir de digitalizações tridimensionais de maneira automatizada, passavam pela necessidade de programação tradicional de algoritmos, como no trabalho de Leal *et al.* (2010), o qual apresenta uma solução que envolve a projeção dos pontos da nuvem obtida pela digitalização sobre uma superfície plana para conversão direta em superfície NURBS, mas a possibilidade de construir algoritmos para tais fins por meio de interfaces eminentemente gráficas, como a proporcionada pelo *plug-in* Grasshopper® do *software* Rhinoceros®, abriu caminho para soluções como a apresentada por Bressan *et al.* (2017), que planifica módulos ou formas poligonais, como em uma folha de papel, para posterior conversão em superfícies NURBS, utilizando tal ferramenta.

Assim sendo, é possível criar soluções que atendam melhor às necessidades de projetistas de produto do que as oferecidas nativamente pelos *softwares* de modelagem. No Rhinoceros®, por exemplo, como apontado por McNeel (2011), é possível converter uma malha para NURBS com um único comando, o *MeshToNurbs*. Neste processo, cada polígono da malha é convertido em uma superfície NURBS. Entretanto, uma malha pode conter centenas, e até milhares, de polígonos que, quando transformados em NURBS, podem elevar em muitas vezes o tamanho do arquivo, podendo comprometer a memória de trabalho do computador, já que os pontos que definem estes polígonos são, em grande parte, os responsáveis por isto. Além disso, uma vez gerado o objeto NURBS, praticamente não é possível manipular e editar esse objeto, em virtude do número de superfícies independentes.

EXPERIMENTO

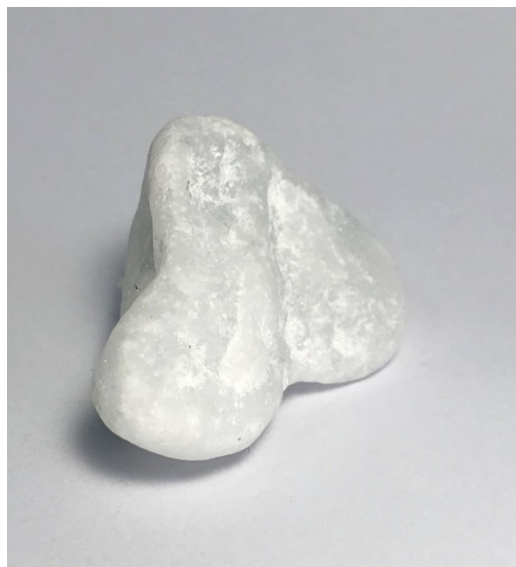
O experimento foi realizado em cinco etapas, apresentadas a seguir. A primeira etapa tratou da escolha do objeto. A segunda, da digitalização 3D da rocha ornamental de forma irregular escolhida. Na terceira etapa, o resultado da digitalização foi convertido em uma malha poligonal. Na quarta, foi desenvolvido um código para conversão de malhas poligonais em superfícies NURBS através do *software* Grasshopper®. Na sequência, foi realizada uma análise da conversão de maneira dimensional, considerando o desvio quadrático médio da malha convertida em relação à malha obtida pela digitalização 3D, por meio do *software* Geomagic Qualify® e, também, uma análise visual por meio da imagem renderizada da malha convertida em relação à malha digitalizada. Por fim, comparou-se o número de vértices da malha poligonal e o número de pontos de controle da superfície NURBS.

ESCOLHA DO OBJETO

A peça escolhida para digitalização foi uma rocha ornamental branca (Figura 1), por conta de sua forma natural irregular com algumas reentrâncias mas sem partes oclusas. Foi, também, relevante, o seu tamanho, aproximadamente, em mm, 41 x 37 x 22, possibilitando a sua colocação no espaço destinado à digitalização pela máquina descrita no passo seguinte.

Figura 1. Peça original

Fonte: dos autores



DIGITALIZAÇÃO 3D

A digitalização 3D da peça foi realizada em um equipamento Digimil 3D, que adquire apenas a forma e não a cor (aspecto não relevante neste experimento). A escolha deste equipamento se deu por conta de sua precisão (0,035 mm), adequada para captura com exatidão necessária do objeto escolhido e pela sua disponibilidade para uso dos autores. Foi usada uma lente de 150 mm e resolução de 0,1 mm. Para obtenção de toda a volumetria da peça foram necessárias seis vistas (Figura 2).

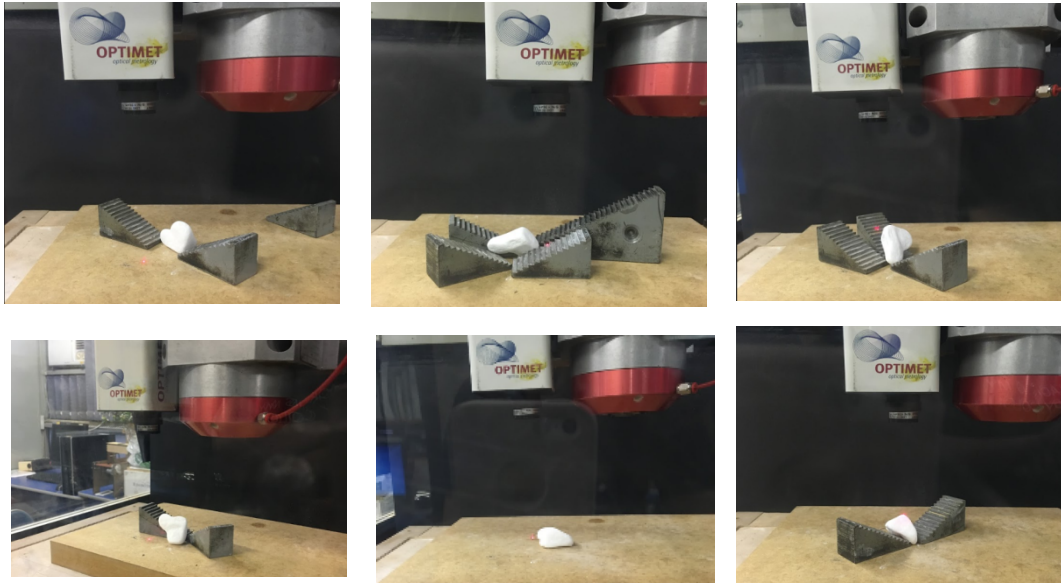


Figura 2.
Posicionamentos da peça
para digitalização

Fonte: dos autores

CONVERSÃO DA NUVEM DE PONTOS EM MALHA POLIGONAL

O processamento das vistas e geração da malha poligonal foi realizada pelo *software* Geomagic Qualify®, no qual foi necessário alinhar todas as vistas, para que tivessem pontos coincidentes, e pudesse ser gerada a malha. A escolha deste programa se deu pela sua disponibilidade de uso para os pesquisadores e pelo controle do processo de conversão das nuvens de pontos em malha poligonal que ele proporciona.

ELABORAÇÃO DO CÓDIGO DE CONVERSÃO

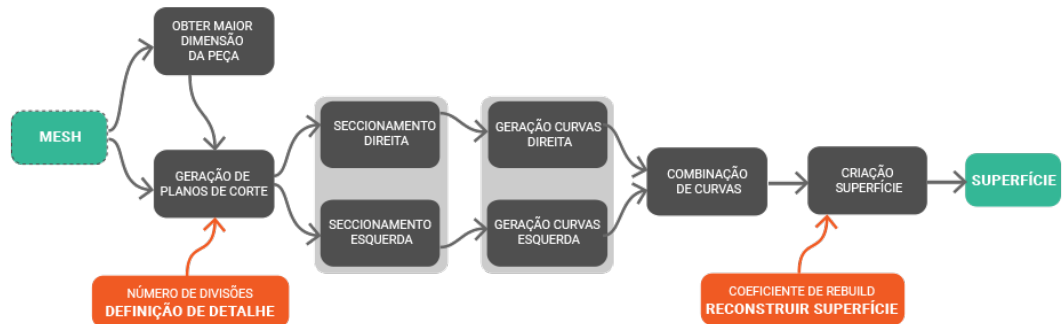
O *software* Rhinoceros® foi escolhido por possibilitar, por meio do *plug-in* Grasshopper®, uma programação visual para construção do algoritmo responsável pela conversão desejada. Desta forma, foi possível trabalhar com dados de entrada (*inputs*) e dados de saída (*outputs*) sem a necessidade de uso de uma linguagem de programação. Tal escolha permite, ainda, que seja possível apresentar de maneira visual o processo de construção do algoritmo, e, dessa maneira, torná-lo mais acessível aos projetistas interessados no seu uso.

Dentre as opções disponibilizadas pelo *software* Rhinoceros® para a construção de superfícies foi escolhido o comando *loft*, pois a estratégia considerada para a elaboração do código consistiu em seccionar o modelo com superfícies planas para a obtenção de curvas que pudessem ser usadas nesta construção, tornando este comando uma das opções mais adequadas.

Desta forma, foram traçadas superfícies planas tendo como ponto de partida o eixo central da peça (de maneira radial). Para isso, o código foi construído em seis etapas: (1) obtenção da maior dimensão da peça; (2) geração de superfícies planas de corte; (3) seccionamento da peça; (4) geração de curvas; (5) combinação de curvas; e (6) criação de superfície a partir do *loft* (Figura 3).

Figura 3. Fluxograma de funcionamento do código final

Fonte: dos autores



A fim de gerar seções que englobassem a peça toda, iniciou-se o código pela obtenção das dimensões da peça nos três eixos (x, y e z). Para isso, o código foi desenvolvido para criar uma caixa (*box*), que envolvesse a peça escaneada e, então, fossem obtidas as medidas da geometria criada. A maior dessas dimensões foi usada como base nas etapas seguintes. Esse processo visou facilitar a readequação do código a outras geometrias, sem a necessidade do *input* manual das dimensões das geometrias a serem convertidas.

Para a geração das superfícies de corte, foi estabelecido que o código obtivesse o centro geométrico da malha, reconhecimento feito a partir do seu volume. O centro geométrico foi obtido a partir da malha original, e não da caixa, pois estes pontos podem não ser equivalentes, considerando-se que o ponto de partida do processo deveria ser a própria malha. Assim, a partir do centro volumétrico, o código traça um círculo considerando como raio a maior dimensão da peça (obtida na etapa anterior), a fim de certificar-se que ele fique maior que a peça. Com o círculo traçado, estipulou-se que o código o dividisse em duas partes. Essa definição foi inevitável pois, com o círculo fechado, o código passava a criar superfícies duplicadas, o que, posteriormente, iria influenciar no ordenamento necessário das curvas para a geração do *loft*. No processo experimental de elaboração do código, quando esta divisão não foi usada, a sequência em que as curvas se apresentavam não possibilitava a execução do *loft*. Tal opção deu-se, também, para que o código pudesse funcionar sem a necessidade de uma reordenamento manual da lista das curvas – maneira como o *software* organiza tais elementos, o que poderia ser necessário na conversão de malhas poligonais de diferentes formatos e prejudicaria a automação do processo.

Considerando-se esta necessidade, conseqüentemente, todas as etapas seguintes ocorreram decompostas em direita e esquerda. Com o círculo dividido em duas partes, estabeleceu-se que as curvas resultantes fossem divididas em pontos igualmente distantes (Figura 4 a).

A quantidade de pontos correspondente ao número de divisões é variável, sendo o primeiro controle livre para ajuste pelo usuário e, portanto, a primeira variação do código. O propósito, de deixar tal controle ajustável, é permitir maior ou menor fidelidade dimensional do objeto convertido em relação à malha original, pois tal nível de fidelidade vai depender das necessidades do projetista. Tendo o eixo central como origem e as divisões do círculo como pontos finais, são geradas linhas distribuídas radialmente (Figura 4 b).

Estas linhas radiais são traçadas a partir do centro geométrico da peça e, por isso, é necessário que elas sejam deslocadas para a base da peça, a fim de as englobar na sua totalidade. Para esse deslocamento é utilizada a maior dimensão da peça, obtida no início do código. O código extruda essas linhas na direção z, permitindo, por fim, a divisão da peça de acordo com as superfícies planas definidas (Figura 4 c). O valor da extrusão é duas vezes a maior dimensão da peça, a fim de garantir que toda peça seja englobada.

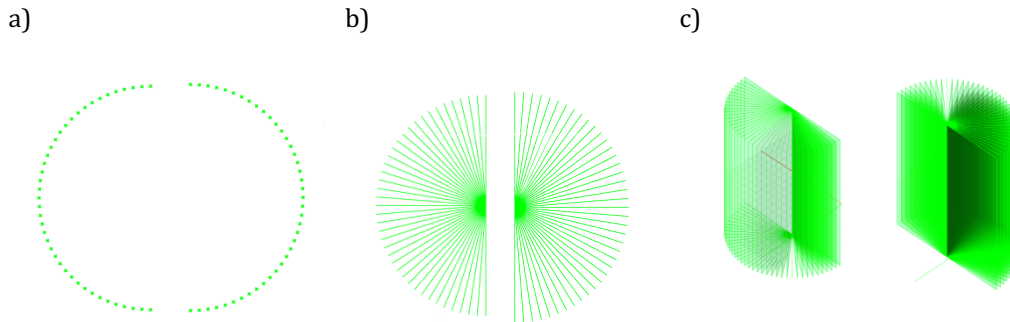


Figura 4. a) divisão dos círculos em pontos, b) linhas radiais, c) superfícies correspondentes às linhas extrudadas

Fonte: dos autores

A etapa seguinte é o seccionamento da peça de acordo com as superfícies. A malha é seccionada, a partir da mesma superfície que dividiu o círculo e, então, passa a ser dividida conforme as extrusões, isto é, cada porção, direita e esquerda, conforme suas superfícies correspondentes. O seccionamento gera diversas curvas polilinhas (*polylines*), que são transformadas em superfície editável ao final do processo (Figura 5). Essa etapa é realizada a partir do comando *loft* e, para isso, são necessárias curvas abertas. Dessa forma, foi necessário estipular que o código não considerasse as curvas fechadas, somente as abertas.

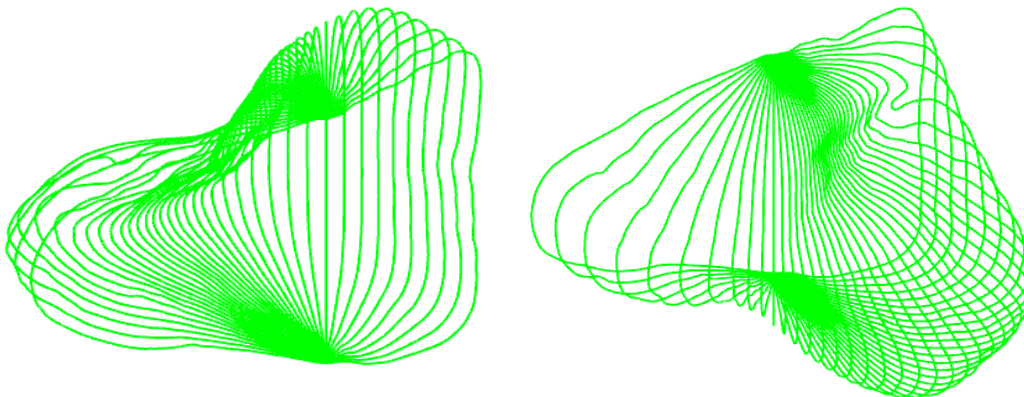


Figura 4. a) divisão dos círculos em pontos, b) linhas radiais, c) superfícies correspondentes às linhas extrudadas

Fonte: dos autores

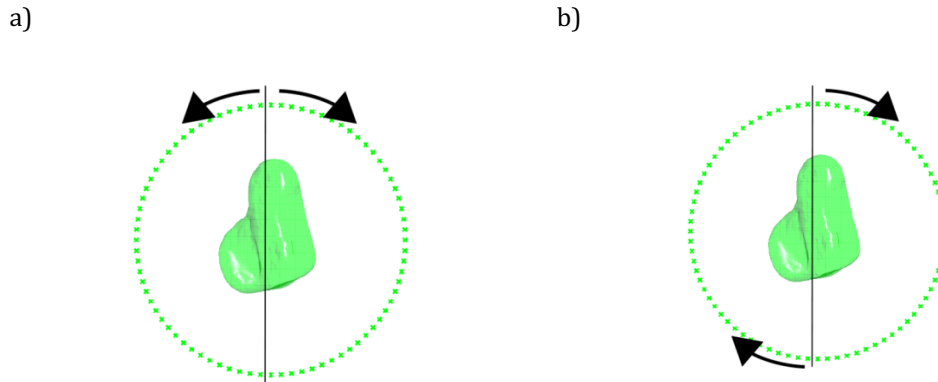
Após a criação das polilinhas, foi necessário utilizar o comando *smooth polyline* (suavizar polilinhas), para que a complexidade destas não sobrecarregasse o processamento de dados. Na sequência, definiu-se que o código transformasse as polilinhas em curvas interpoladas. Para isso, o código explode as polilinhas em pontos para, a partir dos pontos, gerar curvas interpoladas, ou seja, curvas NURBS.

De acordo com a divisão inicial do código, até esta etapa, o processamento acontece em duas partes (direita e esquerda). Por esse motivo, foi necessário estabelecer que o código ordenasse as curvas para que o comando *loft* seguisse em apenas um sentido. Sem esse

ordenamento, o *loft* é iniciado nos dois lados a partir do ponto inicial (Figura 6 a), enquanto é imprescindível que o segundo *loft* siga a partir do ponto final do primeiro (Figura 6 b).

Figura 6. a) *loft* sem ordenamento, b) *loft* com ordenamento

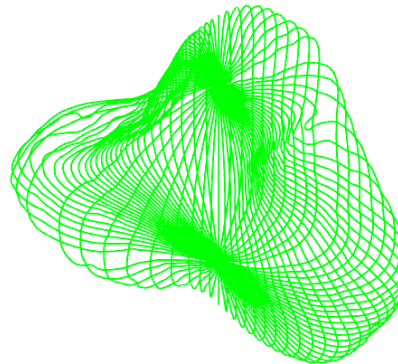
Fonte: dos autores



A etapa seguinte combina as duas partes, direita e esquerda, e, para que o código funcionasse corretamente, foi necessário, primeiro, conectar a parte onde o *loft* deve iniciar e, só então, a segunda parte (Figura 7).

Figura 7. Curvas combinadas

Fonte: dos autores



Com as curvas combinadas, a próxima etapa realiza, efetivamente, o comando *loft*. Ressalta-se que outros comandos poderiam ter sido utilizados acompanhando diferentes perspectivas de modelagem. Além disso, evidencia-se que a escolha considerou o uso de curvas em apenas um sentido, visando minimizar o processamento necessário para a conversão.

Nessa mesma etapa, é aplicado o *rebuild surface* (reconstrução de superfície). Desta forma, é o segundo controle que fica livre para ajuste do usuário e, portanto, a segunda variação possível do código. Este comando divide, uniformemente, a superfície em um número igual de pontos de controle, embora isso tenha impacto na geometria. A possibilidade de edição deste coeficiente permite variar a fidelidade dimensional em relação ao objeto original. Também testou-se a geração de superfície sem a reconstrução, porém, o *software* ficou sobrecarregado e não foi gerada a superfície final.

O código fornece diferentes opções para o comando *loft* – *normal*, *loose*, *tight*, *straight*, *developable* e *uniform* – sendo que a sua escolha dependerá das intenções do projetista (maior ou menor fidelidade ao objeto original).

METODOLOGIA DE ANÁLISE DIMENSIONAL E VISUAL

A comprovação do funcionamento do código, além da obtenção de uma superfície NURBS ao final do processo, necessita de uma análise dimensional e visual para que seja aferida a possibilidade de conversão, com controle de fidelidade, desta superfície em relação a malha poligonal original. Outro ponto a ser considerado no funcionamento do código é a sua capacidade de diminuir o tamanho do arquivo do modelo em virtude da diminuição do número de pontos de controle da superfície NURBS em relação aos vértices da malha poligonal.

Para análise dos resultados, gerou-se dezoito variações, dentro das variáveis disponíveis no código, que representam os pontos de controle pelo usuário. As variáveis foram: segmentos (número de divisões), coeficiente de reconstrução (*rebuild*) e tipo de *loft* (normal ou *loose*).

No processo de desenvolvimento do código, todas as opções foram testadas, mas os modos *tight*, *straight*, *developable* e *uniform* não apresentaram, visualmente, diferenças em relação à opção normal. O *loft* tipo *loose* não é tão rigoroso ao conectar os pontos de controle, sendo similar à diferença entre uma curva interpolada e uma curva de Bézier, onde os pontos são apenas pontos de controle. Consequentemente, o tipo *loose* deixa a peça visualmente mais suave e, aparentemente, com volume menor, enquanto o normal, também aparentemente, não possui perda de volume, porém conta com uma forma visualmente mais grosseira. Assim, as opções de *loft* testadas foram a normal e a *loose*, por serem as opções que mais diferem entre si dentre as disponibilizadas pelo *software*.

Para número de divisões e coeficiente de reconstrução, optou-se por três valores que foram estabelecidos conforme variações de ordem visual significativas na peça, sem que houvesse grande distorção da forma original. Assim, foram estabelecidos os valores de 10, 35 e 70 para as divisões (valores acima de 70 tornaram o processo lento para o computador usado nas análises) e 20, 40 e 80 para o coeficiente de reconstrução.

Cada variação gerada foi carregada no *software* Geomagic Qualify® e sobreposta com o modelo virtual escaneado, ou seja, o modelo de referência para fins de comparação da precisão da superfície NURBS pelo método de análise de desvio (*Deviation Analysis*). Dessa sobreposição dos dois modelos, resulta um desvio, que pode ser observado mais detalhadamente pelo parâmetro de desvio quadrático médio (RMS). Com base na escala de cores, é possível observar no modelo o mapa de cores do desvio resultante (Tabela 1).

O RMS representa a diferença dimensional do modelo NURBS em relação ao modelo 3D de referência escaneado. As partes da peça, representadas com a cor cinza, são os locais nos quais o *software* não encontrou referências para realizar a medição da peça e, portanto, não foi capaz de gerar um desvio aparente. Quanto mais a cor verde estiver presente no modelo, mais próximo do original o modelo convertido é. As partes na cor vermelha são porções externas ao modelo escaneado e, as em azul, são internas.

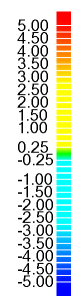
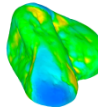
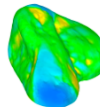
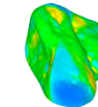
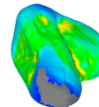
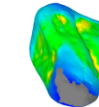
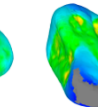






Além disso, o *software* faz a comparação a partir da tolerância configurada na escala da análise, que nesse caso foi de 0,25 mm. Na análise visual das peças, percebeu-se algumas áreas com distorções na aparência das conversões em relação ao modelo original que não ficaram evidentes na análise dimensional. Isso se deve, possivelmente, a essas deformações ficarem dentro da tolerância estabelecida, de 0,25 mm. Dessa forma, para que a análise fosse mais completa, considerou-se tanto a análise dimensional quanto a visual das peças.

RESULTADOS

Os resultados da análise mostram que os modelos gerados com o *loft normal* apresentam um erro quase 50% menor do que aqueles gerados pelo *loft loose*. Para os números de divisões adotados, o desvio quadrático médio das diferenças entre a peça original e as peças geradas, para os dois tipos de *loft*, foi semelhante para todos os valores de reconstrução. Tratando-se do *render*, é possível observar, visualmente, que se perderam detalhes da textura da rocha. Tais detalhes podem estar dentro da tolerância da análise e, por isso, não alteraram a escala de cor e são observados apenas na análise visual. A Tabela 1 apresenta os resultados de cada uma das variações geradas pela divisão em 10 segmentos, bem como o *render* do respectivo modelo.

Tabela 1. Modelos construídos a partir de 10 segmentos do modelo original

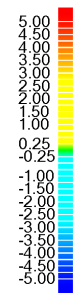
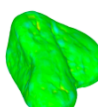
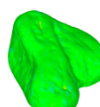
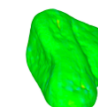
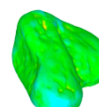
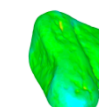
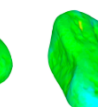






Fonte: dos autores

Modelos com 10 segmentos							
<i>Loft</i>	Normal			<i>Loose</i>			
Reconstrução	20	40	80	20	40	80	
Análise							
Render							
Raiz quadrática média	0,35 mm	0,39 mm	0,38 mm	0,70 mm	0,70 mm	0,69 mm	(mm)

Por sua vez, para os modelos construídos a partir de 35 segmentos, cujos resultados encontram-se na Tabela 2, os erros obtidos foram consideravelmente menores.

Tabela 2. Modelos construídos a partir de 35 segmentos do modelo original

Fonte: dos autores

Modelos com 35 segmentos							
<i>Loft</i>	Normal			<i>Loose</i>			
Reconstrução	20	40	80	20	40	80	
Análise							
Render							
Raiz quadrática média	0,06 mm	0,03 mm	0,03 mm	0,08 mm	0,07 mm	0,06 mm	(mm)

Em linhas gerais, apesar do uso do *loft loose* garantir uma suavização da superfície gerada, os maiores erros ocorreram nas superfícies geradas com este parâmetro, juntamente com o uso do coeficiente de reconstrução em 20, o que pode distanciar o modelo 3D NURBS do modelo de referência. Os maiores erros encontrados foram nas peças elaboradas a partir de 10 segmentos e com *loft loose*.

Comparando-se os modelos, aquele que apresentou maior precisão, em relação ao modelo de referência, foi o construído a partir de 70 segmentos, com *loft* normal e coeficiente de reconstrução igual a 80, cuja raiz quadrática média das diferenças em relação à peça original foi de 0,01 mm, menor erro em comparação aos demais (Tabela 3). Este modelo 3D NURBS apresenta a maior fidelidade dimensional, na amostra, à malha poligonal original. Tais resultados confirmam que, quanto maiores a quantidade de divisões do segmento e o coeficiente de reconstrução, maior a fidelidade dimensional do modelo NURBS. Na Figura 8 pode ser observado o aspecto visual desta fidelidade.

Modelos com 70 segmentos							
<i>Loft</i>	Normal			<i>Loose</i>			
Reconstrução	20	40	80	20	40	80	
Análise							
Render							
Raiz quadrática média	0,07 mm	0,02 mm	0,01 mm	0,06 mm	0,02 mm	0,02 mm	(mm)

Tabela 3. Modelos construídos a partir de 70 segmentos do modelo original, com o melhor resultado da análise destacado em negrito

Fonte: dos autores



Figura 8. Da esquerda para direita, a pedra original, o modelo de malha poligonal e o modelo NURBS mais fiel ao modelo poligonal

Fonte: dos autores

Por fim, comparou-se o número de vértices da malha poligonal, usada na conversão, com os da superfície NURBS com a maior fidelidade dimensional resultante do processo. O modelo tridimensional original apresentou 208.797 pontos, enquanto a superfície convertida com o melhor resultado, 10.766 pontos de controle. Além disso, o arquivo da malha poligonal tem 20 MB de tamanho, frente a 6 MB do arquivo da superfície NURBS.

Como forma de delimitar a abrangência de funcionamento do código, posteriormente às conversões da malha poligonal original segundo os parâmetros descritos neste artigo, foram realizados testes com malhas poligonais de diferentes formas.

Tais testes indicaram que o código não funciona para objetos nos quais as superfícies de corte, que determinam as curvas que irão ser usadas no *loft*, gerem mais de uma curva no mesmo plano.

CONSIDERAÇÕES FINAIS

Apresenta-se, neste artigo, uma proposta de conversão automática de uma malha poligonal, gerada a partir do processamento de uma nuvem de pontos originária de digitalização 3D, em superfícies NURBS editáveis e compatíveis com *softwares* CAD/CAM. O código gerado no Grasshopper® apresenta uma opção ao usuário de transformar malhas poligonais em NURBS obtendo um modelo 3D mais fiel ao modelo original do que utilizando comandos padrões oferecidos em alguns *softwares* e, também, potencialmente, um arquivo de menor tamanho. Essa conversão busca permitir a facilitação da modelagem de formas da natureza, que são assimétricas e orgânicas, permitindo maior exploração dessas para diferentes fins. O código permite adicionar dados de entrada de acordo com a finalidade do projeto, uma vez que se torna possível definir o nível de detalhamento (número de segmentos) e a definição (coeficiente de reconstrução) da superfície que se quer gerar. Dessa forma, caso o objetivo seja gerar um modelo fiel à realidade, para fins de reprodução do objeto original, pode-se utilizar valores maiores desses parâmetros. Caso o objetivo seja utilizar a forma gerada como base para modelagem e criação de diferentes formas, pode-se optar por uma configuração mais simples, que torne o modelo mais fácil de ser manipulado. O código apresenta algumas limitações de uso: formas que, ao serem seccionadas pelas superfícies de corte do código, gerem duas curvas neste mesmo plano em que ocorre o corte, não são possíveis de conversão, as quais poderão ser abordadas em estudos futuros.

Agradecimentos

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de financiamento 001.

Referências Bibliográficas

ANDERSON, C. W.; CRAWFORD-HINES, S. Fast Generation of NURBS Surfaces from Polygonal Mesh Models of Human Anatomy. **Colorado State University Computer Science Technical Report CS-99**, Fort Collins, USA, n. 101, Feb. 2000. Disponível em: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.450.6980&rep=rep1&type=pdf>. Acesso em: 10 set. 2020.

BRESSAN, F.; KINDLEIN JÚNIOR, W.; TEIXEIRA, F. G. Design e Tecnologia: transposição de volumetrias entre superfícies poligonais e *polysurfaces* NURBS. **Educação Gráfica**, v. 21, n. 3, p. 213-229, 2017. Disponível em: <http://www.educacaografica.inf.br/revistas/vol-21-numero-03-2017>. Acesso em: 10 maio 2019.

FOLEY, J. D.; VAN DAM, A.; FEINER, S. K.; HUGHES, J. F.; PHILLIPS, R. L. **Introduction to Computer Graphics**. Boston: Addison-Wesley Publishing Company, 1994.

GROETELAARS, N.; AMORIM, A. 3D Laser Scanning Technology: characteristics, processes and point cloud tools. *In*: IBEROAMERICAN CONGRESS OF DIGITAL GRAPHICS: AUGMENTED CULTURE, 15th, 2011, Santa Fe. **Proceedings...** Santa Fe: Universidad Nacional del Litoral, 2011. p. 490-494. Disponível em: https://cumincad.architexturez.net/system/files/pdf/sigradi2011_125.content.pdf. Acesso em: 10 maio 2019.

KINDLEIN JÚNIOR, W.; GUANABARA, A. S. Methodology for product design based on the study of bionics. **Materials & Design**, v. 26, n. 2, p. 149-155, 2005. Disponível em: <https://www.sciencedirect.com/science/article/abs/pii/S0261306904001153>. Acesso em: 15 dez. 2020.

LEAL, N.; LEAL, E.; BRANCH, J. W. Simple Method for Constructing NURBS Surfaces from Unorganized Points. In: INTERNATIONAL MESHING ROUNDTABLE, 19th, 2010, Chattanooga. **Proceedings...** Berlin: Springer, 2010. p. 161-175. Disponível em: https://doi.org/10.1007/978-3-642-15414-0_10. Acesso em: 10 maio 2019.

PALOMBINI, F. L.; KINDLEIN JÚNIOR, W.; SILVA, F. P.; MARIATH, J. E. A. Design, Biônica e novos paradigmas: uso de tecnologias 3D para análise e caracterização aplicadas em anatomia vegetal. **Design e Tecnologia**, v. 7, n. 13, p. 46-56, 2017. Disponível em: <https://lume.ufrgs.br/bitstream/handle/10183/169911/001051734.pdf?sequence=1&isAllowed=y>. Acesso em: 22 maio 2019.

PALOMBINI, F. L.; PESTANO, V.; KINDLEIN JÚNIOR, W.; CUNHA DUARTE, L. Biônica e Seleção de Materiais Celulares para projetos de Design. **Design e Tecnologia**, v. 10, n. 20, p. 01-10, 2020. Disponível em: <https://www.ufrgs.br/det/index.php/det/article/view/558>. Acesso em: 15 dez. 2020.

McNEEL, R. The MeshToNurb Command. **McNeel Wiki**, 13th Apr. 2011. Disponível em: <https://wiki.mcneel.com/rhino/meshtonurb>. Acesso em: 24 maio 2019.

QUEIROZ, N.; RATTES, R.; BARBOSA, R. Biônica e biomimética no contexto da complexidade e sustentabilidade em projeto. In: ARRUDA, A. J. V. (org.). **Design & Complexidade**. São Paulo: Editora Blucher, 2017. p. 125-140.

TEDESCHI, A. **Parametric Architecture with Grasshopper**. Brienza: Le Penseur, 2011.

Marina Pires de Castro Aguiar
Vale

marinapires.design@gmail.com

Camila Vieira Ghisleni
mila.ghisleni@gmail.com

Fernanda Fontana de Gasperin
fernanda.gasperin@gmail.com

Roger Langone Leal
rogerlangone@hotmail.com

Fabio Pinto da Silva
fabio.silva@ufrgs.br

Underléa Miotto Bruscato
underlea.bruscato@ufrgs.br