



Edição de gcodes e manipulação da impressão 3d.

João V. L. Capi*, Varlei Rodrigues, Antônio Riul Jr.

Resumo

Neste projeto pretendíamos desenvolver um programa em python para modificar de forma sistemática o conjunto de instruções enviadas a uma impressora 3D, na impressão de dispositivos de microfluídica, para aumentar o controle que tínhamos da trajetória que o bico de impressão faz ao depositar o material de impressão.

Palavras-chave:

gcode, impressão 3d, microfluídica

Introdução

A microfluídica trabalha com a manipulação de fluidos em canais com dimensões submilimétricas, possibilitando o desenvolvimento de sistemas miniaturizados de análise. A impressão 3D destes dispositivos permite a redução de custos e de etapas de processamento. Na impressão 3D, usamos o método *Fused Deposition Modeling*, em que um termoplástico é extrusado a quente e então depositado sobre uma mesa, camada a camada. Um código gcode fornece os parâmetros para a impressão e a sequência de movimentos que imprimem a peça tridimensionalmente. Esse código é gerado a partir da geometria da superfície de um objeto em três dimensões. Um software fatiador corta o modelo 3D em camadas 2D, obtendo as coordenadas a serem preenchidas e a quantidade de termoplástico a ser depositado. Um dos problemas identificados na impressão de microcanais é que os fatiadores em geral são voltados para a prototipagem de modelos muito maiores que os nossos e traçam os caminhos da impressão visando minimizar o tempo. Em particular, para a redução do tempo de impressão os programas minimizam as trajetórias seguidas pelo bico de impressão, o que frequentemente passa pelo microcanal e o deforma. O objetivo desse projeto foi desenvolver um programa em python para modificar de forma sistemática o arquivo gcode usado na impressão de dispositivos de microfluídica e aumentar o controle que temos da trajetória que o bico faz ao depositar o material de impressão.

Resultados e Discussão

Para o desenvolvimento do projeto era preciso inicialmente aprender a desenhar peças 3D, a fatiar o objeto e obter o gcode, aprender a linguagem python, modificar o gcode e testar o novo código em uma impressão real. Para isso, começamos com o desenvolvimento das seguintes tarefas básicas:

- Função *Merge*: Une dois arquivos gcode em um único e os imprime simultaneamente, camada à camada.
- Função *Upper Translation*: Une dois arquivos gcode em um único e imprime o segundo sobre o primeiro.

- Função *Modified Upper Translation*: Une dois arquivos gcode em um único e imprime o segundo sobre o primeiro. Nesta função ocorre uma pausa entre a impressão das peças para troca do termoplástico.

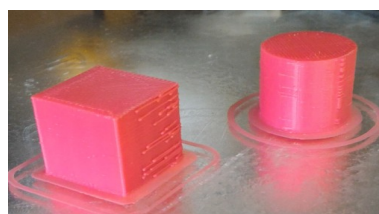


Figura 1. Resultado da função *Merge* de um cubo e um cilindro, gerados originalmente em gcodes separados.

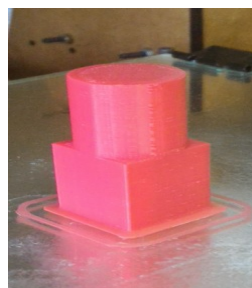


Figura 2. Resultado da função *Upper Translation* de um cubo e um cilindro gerados originalmente em gcodes separados.

Conclusão

Concluimos que esses códigos automatizarão processos que são feitos manualmente por outros membros em nosso grupo de pesquisa, aumentando o controle da trajetória que o bico faz ao depositar o termoplástico e prevenindo danos na impressão de microcanais.

Agradecimentos

Agradeço ao *Programa Institucional de Bolsas de Iniciação Científica*, pelo fomento à essa pesquisa, a Marcos L. F. Gomes pela ajuda com a impressão dos testes e a Gabriel Gaal pela ajuda com os comandos gcode.

Y. He, Y. Wu, J. Fu, Q. Gao e J. Qiu, Developments of 3D Printing Microfluidics and Applications in Chemistry and Biology: a Review, *Electroanalysis*, **2016**, 28, 1658 - 1678.
X. Wang, M. Jiang, Z. Zhou, J. Gou e D. Hui, 3D printing of polymer matrix composites: A review and prospective, *Composites Part B*, **2017**, 110, 442 - 458.
S. Waheed, J. M. Cabot, N. P. Macdonald, T. Lewis, R. M. Guijt, B. Paull e M. C. Breadmore, 3D printed microfluidic devices: enablers and barriers, *Lab on a Chip*, **2016**, 16, 1993 - 2013.