



UNIVERSIDADE DE BRASÍLIA – UnB
FACULDADE DE ARQUITETURA E URBANISMO – FAU
PROGRAMA DE PÓS-GRADUAÇÃO EM ARQUITETURA E URBANISMO – PPG-FAU
DOUTORADO EM ARQUITETURA E URBANISMO

**ALGORITMOS E REQUISITOS PARA DEFINIÇÃO DE FERRAMENTA DE
CÓDIGO ABERTO PARA IMPRESSÃO 3D DE ARGAMASSAS**

*Algorithms and Requirements for defining an open-source tool for 3D printing
of mortars*

Renan do Nascimento Balzani

Orientador: Prof. Dr. Neander Furtado Silva

Brasília/DF

2021



UNIVERSIDADE DE BRASÍLIA – UnB
FACULDADE DE ARQUITETURA E URBANISMO – FAU
PROGRAMA DE PÓS-GRADUAÇÃO EM ARQUITETURA E URBANISMO – PPG-FAU
DOUTORADO EM ARQUITETURA E URBANISMO

**ALGORITMOS E REQUISITOS PARA DEFINIÇÃO DE FERRAMENTA DE
CÓDIGO ABERTO PARA IMPRESSÃO 3D DE ARGAMASSAS**

*Algorithms and Requirements for defining an open-source tool for 3D printing
of mortars*

RENAN DO NASCIMENTO BALZANI

Tese apresentada ao Programa de Pós-Graduação da Faculdade de Arquitetura e Urbanismo da Universidade de Brasília como requisito parcial à obtenção do título de DOUTOR em Arquitetura e Urbanismo.

Área de concentração: Tecnologia, Ambiente e Sustentabilidade (TAS).

Orientador: Prof. Dr. Neander Furtado Silva

Brasília/DF

2021

RENAN DO NASCIMENTO BALZANI

**ALGORITMOS E REQUISITOS PARA DEFINIÇÃO DE FERRAMENTA DE
CÓDIGO ABERTO PARA IMPRESSÃO 3D DE ARGAMASSAS**

Tese apresentada ao Programa de Pós-Graduação da Faculdade de Arquitetura e Urbanismo da Universidade de Brasília como requisito parcial à obtenção do título de DOUTOR em Arquitetura e Urbanismo.

BANCA EXAMINADORA

Prof. Dr. Neander Furtado Silva (PPG-FAU/UnB)
(Orientador e Presidente)

Prof. Dr. Márcio Augusto Roma Buzar (PPG-FAU/UnB)
(Examinador Interno)

Prof. Dr. Igor Lacroix (Universidade do Porto, Portugal)
(Examinador Externo)

Prof. Dr. Juan Carlos Guillen Salas (FAV/UFG)
(Examinador Externo)

FICHA CATALOGRÁFICA

Ficha catalográfica elaborada automaticamente,
com os dados fornecidos pelo(a) autor(a)

Ba	Balzani, Renan do Nascimento
	ALGORITMOS E REQUISITOS PARA DEFINIÇÃO DE FERRAMENTA DE CÓDIGO ABERTO PARA IMPRESSÃO 3D DE ARGAMASSAS / Renan do Nascimento Balzani; orientador Dr. Neander Furtado Silva. - Brasília, 2021. 193p.
	1. Algoritmo. 2. Código Aberto. 3. Construção Aditiva. 4. Impressão 3D. 5. Impressão de argamassa. I. Furtado Silva, Dr. Neander , orient. II. Título.

REFERÊNCIA BIBLIOGRÁFICA

BALZANI, R. (2021). *Algoritmos e Requisitos para definição de ferramenta de código aberto para impressão 3D de argamassas*. Tese de Doutorado em Arquitetura e Urbanismo. Programa de Pós-Graduação em Arquitetura e Urbanismo, Faculdade de Arquitetura e Urbanismo, Universidade de Brasília, Brasília, DF, 193p.

CESSÃO DE DIREITO

AUTOR: Renan do Nascimento Balzani

CURRÍCULO LATTES: <http://lattes.cnpq.br/4419728092677611>

TÍTULO: Algoritmos e Requisitos para definição de ferramenta de código aberto para impressão 3D de argamassas.

GRAU: Doutor ANO: 2021

É concedida à Universidade de Brasília permissão para disponibilizar cópias desta tese de doutorado e para emprestar ou vender tais cópias somente para propósitos acadêmicos e científicos. O autor reserva outros direitos de publicação e nenhuma parte desta tese de doutorado pode ser reproduzida sem autorização por escrito do autor.

Dedico este trabalho à minha família e amigos por me incentivarem e apoiarem em todos os momentos.

Aos meus amigos e companheiros do curso de Arquitetura e Urbanismo, Ana Luísa, Larissa César e Leonardo Barreto, que me ajudaram com muita paciência e muitas horas de conversas e reflexões.

AGRADECIMENTOS

Agradeço à Universidade de Brasília pela oportunidade de fazer um Doutorado que conjuga arquitetura e tecnologia e crescer como pessoa, estudante e profissional.

Ao meu orientador, professor Dr. Neander Furtado Silva, pelo cuidado constante e paciência. Obrigado pelas orientações para encontrar o tema adequado para o meu perfil de trabalho.

Ao professor Dr. Márcio Albuquerque Buson, por propor o projeto e construção da impressora 3D apresentada no estudo e pelas longas orientações e discussões sobre arquitetura de terra.

Ao meu grande amigo Dr. Félix Alves, por sempre me incentivar a começar o meu doutoramento o mais cedo possível e por me ajudar a organizar minhas ideias.

À minha amiga e professora Dra. Maribel Del Carmen Aliaga Fuentes, pelas longas orientações e revisões sempre pacientes e sensatas.

Ao meu amigo Me. Leonardo Campos Barreto, companheiro de graduação e de mestrado, pelas conversas sobre projeto de impressoras 3D, programação de computadores e arquitetura.

À minha amiga Arqt^a. Larissa César, pelas longas conversas sobre a profissão e sobre o doutorado. Obrigado por me ajudar a manter o foco e a determinação.

À minha amiga Arqt^a. Ana Luísa, por revisar meu trabalho com dedicação e carinho.

Ao meu grande amigo Ewerton Cavalcante, pela ajuda com conceitos e referências de programação e linguagem de computação.

Aos meus pais, Diones Balzani e Leopoldina Rita do Nascimento Balzani, pelo apoio incondicional em todos esses anos de estudo e pesquisa.

RESUMO

Esta pesquisa se encaixa na área de desenvolvimento de soluções tecnológicas para a automação da construção com o uso de impressão tridimensional aditiva por extrusão de argamassa. Apresenta como problemática a definição de parâmetros e algoritmos para impressão 3D de materiais pastosos. Acredita-se que a lacuna no conhecimento pode ser preenchida por meio do desenvolvimento e disponibilização de algoritmos e requisitos para impressão 3D em tamanho natural de materiais pastosos. O estudo tem por objetivo principal apresentar algoritmos de código aberto para definição de impressão 3D com argamassas. Para tal, serão utilizadas estratégias de desenvolvimentos de *softwares* como o *reuse-oriented software engineering*, método amplamente utilizado na criação de ferramentas digitais com códigos abertos. Os parâmetros e algoritmos de impressão vinculados aos aspectos construtivos com argamassas são apresentados em “português estruturado” (Portugol) e fluxogramas de programação. É apresentada na tese a colaboração para o desenvolvimento de projeto e construção de uma impressora 3D para simulação de impressões em tamanho natural. O resultado é a entrega de algoritmos de código aberto para contribuir na elaboração de novos processos construtivos e no aumento do acesso às tecnologias de construção digital, bem como para uso tanto pela construção civil quanto por instituições de ensino e pesquisa.

PALAVRAS-CHAVES: Algoritmo; Código Aberto; Construção Aditiva; Impressão 3D; Impressão de argamassa.

ABSTRACT

This is a research into the area of development of technological solutions for building automation with the use of additive three-dimensional printing by mortar extrusion. It presents as research problem the definition of parameters and algorithms for 3D printing of pasty materials. It is believed that the knowledge gap can be filled by developing and making available algorithms and requirements for full-size 3D printing of soft materials. The main objective of the study is to present open source algorithms for defining 3D printing with mortars. For this purpose, software development strategies such as reuse-oriented software engineering will be used, a method widely used in the creation of digital tools with open codes. The printing parameters and algorithms linked to the construction aspects with mortars are presented in “structured Portuguese” (Portugol) and programming flowcharts. Collaboration for the development of a project and construction of a 3D printer for simulating full-size prints is presented in the thesis. The result is the delivery of open source algorithms to contribute to the development of new construction processes and increase access to digital construction technologies, as well as for use in civil construction and by educational and research institutions.

KEYWORDS: Algorithm; Open Source; Additive Construction; 3D printing; Mortar printing.

Sumário

PREFÁCIO	14
FLUXOGRAMA DE DESENVOLVIMENTO DO TRABALHO.....	17
Introdução	18
1 Contextualização	21
1.1 Questão norteadora	21
1.2 Hipótese	21
1.3 Objetivo	21
1.4 Objetivos específicos	21
1.4.1 <i>Desenvolver blocos de algoritmos</i>	22
1.4.2 <i>Parâmetros das argamassas</i>	22
1.4.3 <i>Desenvolvimento de máquina</i>	23
1.5 Procedimentos metodológicos	23
2 Revisão bibliográfica	24
2.1 Construções de terra.	24
2.2 Evolução da fabricação	26
2.3 A Quarta Revolução Industrial.....	27
2.4 A Indústria 4.0	28
2.5 O plano estratégico da manufatura avançada nos EUA.....	33
2.6 O Japão diante da Indústria 4.0	33
2.7 China e o plano estratégico <i>Made in China 2025</i>	36
2.8 Agenda brasileira para a Indústria 4.0	37
2.9 Fabricação e construção digital aditiva.....	41
2.10 Uso da tecnologia de fabricação e sua evolução para impressão tridimensional de edificações	48
2.11 Impressões 3D de Edificações.....	52
2.12 Traços utilizados.....	51
2.13 Custos da impressão em m ²	52
2.14 Uso da tecnologia de fabricação no ensino de projeto e sua evolução	56
3 Impressora 3D de extrusão de argamassa – PPG-FAU/UnB.....	60
3.1 Programação da Impressora – PPG-FAU/UnB.....	62
3.1.1 <i>Fluxograma para desenvolvimento da programação da impressora</i>	62
3.2 Parâmetros para impressão 3D de argamassa.	63
3.2.1 <i>Blocos de algoritmos para definição de requisitos de impressão 3D de argamassa</i>	64
4 Programação e parâmetros – <i>softwares open-source</i> /livres	66
5 Programas <i>open-source</i> para análise	69
5.1 Algoritmos e requisitos para definição de ferramenta de impressão 3D	71
6 Programa <i>PRONTERFACE/PRINTRUN</i> – requisitos para especificação de <i>software</i>	78
6.1 Blocos para modificação de medidas.....	80
6.1.1 <i>Bloco para medida e unidade da base – “build dimensions”</i>	80
6.1.2 <i>Bloco de algoritmo para medidas e parâmetros de controlador virtual</i>	83

6.1.3 Bloco para definição de diâmetro do bico extrusor	87
6.1.4 Definição de algoritmo de pausa e retomada de impressão.....	90
6.2 Algoritmo de controle de velocidade	93
6.2.1 Bloco de algoritmo para controle de velocidade de movimento dos eixos sem extrusão.....	94
7 Programa Slic3r – requisitos de especificação de software	98
7.1 Algoritmo para volume de impressão em metros.....	101
7.2 Algoritmo relacionado à retração da argamassa para compensar possível deformação das camadas.....	103
7.3 Algoritmo relacionado à plasticidade das argamassas para definir a velocidade de extrusão.....	106
7.4 Requisitos para modificação dos parâmetros de temperatura.....	109
7.5 Bloco de algoritmo para definição dos perímetros das linhas de impressão vinculado ao tamanho do bico extrusor.....	111
7.5.1 Layer height (altura de camada).....	112
7.5.2 Vertical shells (perímetros verticais)	114
7.5.3 Horizontal Shells (preenchimentos horizontais).....	115
7.5.4 Comandos Quality (slower slicing) – Qualidade (fatiamento lento) e Advanced (avançado).....	116
7.5.5 Comandos Infill (preenchimento).....	117
7.5.6 Comandos Speed (velocidade).....	125
7.5.7 Comandos Custom G-code (Código G personalizado).....	127
7.5.8 Start G-code definido para impressora FAU/UnB:.....	128
7.5.9 End G-code definido para impressora FAU/UnB:.....	129
7.5.10 Between objects for sequential printing (entre impressão sequencial). em formato código G (Gcode);.....	130
8 Resultados.....	132
9 Conclusão.....	157
10 Estudos futuros e recomendações.....	163
Referências.....	165
APÊNDICE A - FLUXOGRAMA Algoritmo BASE_METROS.....	171
APÊNDICE B - FLUXOGRAMA Algoritmo MOVIMENTO_Z.....	172
APÊNDICE C - FLUXOGRAMA Algoritmo MOVIMENTO_XY.....	173
APÊNDICE D - FLUXOGRAMA Algoritmo BICO_EXTRUSAO.....	174
APÊNDICE E - FLUXOGRAMA Algoritmo PAUSA_RETOMADA.....	175
APÊNDICE F - FLUXOGRAMA Algoritmo VELOCIDADE_SEM_EXTRUSAO.....	176
APÊNDICE G - FLUXOGRAMA Algoritmo VOLUME_METROS.....	178
APÊNDICE H - FLUXOGRAMA Algoritmo RETRAÇÃO.....	180
APÊNDICE I - FLUXOGRAMA Algoritmo PLASTICIDADE_VELOCIDADE.....	182
APÊNDICE J - FLUXOGRAMA Algoritmo PRIMEIRA_CAMADA.....	184
APÊNDICE K - FLUXOGRAMA Algoritmo ZIGUE_ZAGUE.....	185
APÊNDICE L - FLUXOGRAMA Algoritmo GRADE.....	186
APÊNDICE M - Programação da impressora 3D de extrusão de argamassa – PPG- FAU/UnB.....	187

APÉNDICE N - Exemplo código G comentado.....	191
---	------------

Lista de Figuras

Figura 1 - Impressora tridimensional projetada e construída na FAU/UnB. Grupo de pesquisa do laboratório LFDC.	18
Figura 2 - Edificação de superadobe.	26
Figura 3 - Esquema ilustrativo dos 9 pilares da Indústria 4.0.	30
Figura 4 - Robô Atlas.....	33
Figura 5 - <i>Oculus by</i> Facebook. Realidade Aumentada.	32
Figura 6 - Armazém de distribuição da Amazon.....	32
Figura 7 - Evolução das sociedades em 5 fases.....	35
Figura 8 - Objetivos das iniciativas nacionais do MIC 2025.....	37
Figura 9 - Participação do setor de transformação industrial no PIB (%) 1985 – 2016.....	38
Figura 10 - Índice global de inovação: países mais inovadores.....	39
Figura 11 - Entre 2006 - 2016, a produtividade da indústria brasileira caiu mais de 7 pontos.	39
Figura 12 - Índice global de competitividade da manufatura.....	40
Figura 13 - Impressoras 3D de baixo custo RepRap. Projeto Adrian Bower.....	41
Figura 14 - Digital Grotesque, Suíça, 2013.....	42
Figura 15 - Proposta para estação Lunar, Foster + Partners.	42
Figura 16 - Projeto Glass I.....	43
Figura 17 - Camadas de impressão 3D em concreto.....	44
Figura 18 - Interface de <i>software Potterware</i>	45
Figura 19 - Ilustração da impressora 3D da Contour Crafting.....	49
Figura 20 - Edificação com elementos impressos. ETH de Zurique.....	50
Figura 21 - Projeto fatiado para impressão 3D.	51
Figura 22 - Escritório do Futuro.....	53
Figura 23 - Apis Cor. Conjunto de casas impressas.	54
Figura 24 - Casas desenvolvidas pela empresa chinesa Winsun, em Suzhou, China, 2015.	55
Figura 25 - Primeira casa impressa no Brasil. Universidade Potiguar, RN.....	56
Figura 26 - <i>Lab. Maker/maquetaria</i> . Instituição de ensino Superior IESB. Brasília - DF, 2020.....	57
Figura 27 - FABLAB. MIT <i>Media Lab</i>	58
Figura 28 - Produção de prótese para crianças - estudante de arquitetura e voluntária do projeto e-Nable, IESB.....	59
Figura 29 - Projeto para impressora em escala para argamassa de terra.....	60
Figura 30 - Construção de impressora em escala para argamassa de terra.....	61
Figura 31 - Processo de desenvolvimento de protótipo.	66
Figura 32 - Simulação de impressão 3D - Programação G-code.	67
Figura 33 - Programa de código aberto <i>Slic3r</i>	70
Figura 34 - Modelo de desenvolvimento de <i>softwares Waterfall</i>	71
Figura 35 - Fluxograma, português estruturado e programa na linguagem C.....	73
Figura 36 - Principais símbolos do fluxograma.....	74
Figura 37 - Tela de comando da impressora projetada e construída na FAU/UnB.....	76
Figura 38 - Algoritmo para apresentar na tela LCD a mensagem 'Pos-UnB. 08.06'. Definido em 08/06/2020.....	76
Figura 39 - Interface de uso do programa <i>Printrun/Pronterface</i>	79
Figura 40 - Controlador virtual de eixos e base de impressão em milímetros.	80

Figura 41 - <i>Printrun/Pronterface</i> . Definição de tamanho de área de impressão.....	81
Figura 42 - <i>Slic3r</i> . <i>Menu</i> de configuração do tamanho da base de impressão.....	83
Figura 43 - Controlador virtual. Eixos X, Y e Z.	84
Figura 44 - Definições de espessuras de cada do <i>software</i> proprietário <i>ULTIMAKER</i> Cura.	87
Figura 45 - Bico de impressão fixo – 1 cm de diâmetro. Impressora produzida na FAU/UnB.	88
Figura 46 - Camadas com ondulações e marolas. Impressora de barro, 3D Wasp, Itália	88
Figura 47 - <i>Menu</i> do programa <i>Printrun/Pronterface</i> para escolha de arquivos, início de impressão e pausa	90
Figura 48 - Ícones de controle de velocidade para impressoras <i>desktop</i> para impressão em plástico	94
Figura 49 - <i>Menu file</i> . Programa <i>Slic3r</i>	98
Figura 50 - Licença de código aberto para uso do Firmware Marlin para impressoras 3D....	99
Figura 51 - <i>Software</i> independente <i>Slic3r</i>	100
Figura 52 - Vista superior da base de impressão do programa <i>Slic3r</i>	101
Figura 53 - <i>Menu</i> para definição de alturas de camadas.	104
Figura 54 - <i>Menu</i> para definição de diâmetro de filamento em impressoras FDM.....	107
Figura 55 - Filamento Buzzed. Produzido com resíduos da fabricação de cerveja.....	110
Figura 56 - <i>Slic3r</i> . <i>Menu</i> de parâmetros de retração	111
Figura 57 - <i>Slic3r</i> . Configurações de impressão.	112
Figura 58 - Bico de extrusão feito com impressora 3D FDM para impressão de argamassa.	113
Figura 59 - Configuração de impressão: <i>Vertical Shells</i> (perímetros verticais).	115
Figura 60 - Exemplo de parede sem preenchimento e totalmente preenchido com perímetros	115
Figura 61 - Configuração de impressão: <i>Horizontal Shells</i> (preenchimentos horizontais) ...	116
Figura 62 - <i>Slic3r</i> . <i>Quality (slower slicing)</i> – Qualidade (fatiamento lento) e <i>Advanced</i> (avançado).....	116
Figura 63 - Exemplos de padrões de preenchimentos para impressoras tipo <i>desktop</i>	117
Figura 64 - <i>Slic3r</i> . Comandos de preenchimento	118
Figura 65 - Pontes impressas em impressoras tipo <i>desktop</i>	118
Figura 66 - Padrão de perímetro e preenchimento em zigue-zague.	119
Figura 67 - Gráfico de exemplo para preenchimento em zigue-zague	122
Figura 68 - Padrão de preenchimento em grade. Espaçamento entre camada.....	122
Figura 69 - Padrão de preenchimento em grade com inclinação de 45°	123
Figura 70 - Gráfico de exemplo para preenchimento em grade.....	125
Figura 71 - <i>Slic3r</i> . Definição de parâmetros para controle de velocidade dos eixos	127
Figura 72 - Comandos para definição de código G personalizado	128
Figura 73 - Código G - Pausa para cura. Inserção na ferramenta <i>Slic3r</i>	131
Figura 74 - Programação feita para impressora construída nos laboratórios EdiQuali/LFDC da FAU/UnB.....	187
Figura 75 - Microcontroladora A4988 para motor de passo bipolar.	188
Figura 76 - Eixos de movimento da impressora 3D. Eixo Z tem sua origem/fim de movimento na parte superior do pórtico.....	189
Figura 77 - Simulação de impressão de parede com preenchimento em zigue-zague.....	191

PREFÁCIO

Entre no curso de Arquitetura e Urbanismo da FAU/UnB em 2007, e fiquei impressionado com a maneira de os projetistas utilizarem a tecnologia para desenvolver propostas arquitetônicas. O uso dos instrumentos de desenho técnico, computador e as máquinas de construção digital são tecnologias fascinantes.

O desenho, no sentido de projeto, tanto à mão quanto assistido por computador, nas suas mais diversas ramificações, me motivaram a escolher o campo de estudo e especialização da construção digital e arquitetura algorítmica.

Fui apresentado à tecnologia de impressão 3D no meu segundo ano de graduação, em 2008, pelo professor Neander Furtado. Contudo, as impressoras tridimensionais eram muito caras. Em 2012, conheci o projeto RepRap, tecnologia de código aberto para pessoas comuns construir suas próprias impressoras 3D. Imediatamente, comecei a construir uma impressora e a estudar como utilizar a técnica para desenvolver propostas de arquitetura, urbanismo e produtos.

O processo de construir impressoras 3D melhorou meus conhecimentos de eletrônica, fabricação digital e programação de computadores.

As pesquisas sobre impressão e construção evidenciam que as máquinas e computadores não substituem o intelecto humano e a capacidade de desenvolver propostas inovadoras. É necessária a produção de estudos para incorporar novas técnicas ao ensino e desenvolvimento de projetos arquitetônicos. Esta tese apresenta algoritmos e requisitos de impressão com a intenção de colaborar para o avanço da tecnologia de impressão 3D em tamanho natural e ampliar as pesquisas.

Os métodos de construção computadorizados não diminuem a qualidade nem a criatividade dos projetistas. Acredito que se Le Corbusier (1887-1965) estivesse vivo hoje, seria tão fascinado com a tecnologia de manufatura aditiva como foi com as máquinas de sua época, ou seja, automóveis, aviões e transatlânticos. Certamente, ele utilizaria algoritmos e ferramentas digitais para produzir propostas arquitetônicas inovadoras e cidades futuristas.

Assim, visando maior clareza na organização do trabalho, apresento a estrutura da tese, que está dividida em tópicos, nove ao total, os quais abordam os temas discutidos do seguinte modo:

- **Introdução:** a introdução aborda a importância da tecnologia de impressão 3D para o desenvolvimento de projetos tanto em impressoras de mesa ou em tamanho natural. O tópico apresenta a possibilidade de desenvolver propostas de máquinas de impressão 3D e ferramentas digitais com tecnologia livre (*open-source*);

- **Contextualização:** aqui, é apresentada a questão norteadora do trabalho, assim como a hipótese, objetivos e procedimentos metodológicos. Esse item evidencia como o trabalho foi desenvolvido e os tipos de materiais e ideias abordadas;

- **Revisão bibliográfica:** a revisão propõe uma breve leitura sobre a história das edificações construídas com terra e como a revolução industrial está avançando para a sua quarta fase, denominada Indústria 4.0. O texto apresenta os avanços da indústria e dos métodos de impressão tridimensional em diferentes modelos e funcionalidades como a impressão com argamassa de concreto e o uso da tecnologia para o ensino e para a melhoria da vida das pessoas;

- **Impressora 3D de extrusão de argamassa – PPG-FAU/UnB:** o estudo apresenta o desenvolvimento de uma impressora de argamassa para a execução de testes de movimentação, programação de máquina e para os algoritmos definidos para impressão 3D;

- **Programação e parâmetros – Software *open-source*/livre:** a seção apresenta os principais aspectos necessários para o desenvolvimento de algoritmos e requisitos para a impressão com argamassa em tamanho natural;

- **Programas *open-source* para modificação:** o tópico apresenta os programas *open-source*, códigos liberados para o uso, parâmetros e métodos para o desenvolvimento de ferramentas digitais. A seção demonstra o desenvolvimento de algoritmos por meio do “português estruturado”, Portugol, e fluxogramas de programação;

- **Programa *Pronterface/Printrun* – requisitos de especificação de *software*:** essa seção apresenta o funcionamento da ferramenta e os motivos de ter sido escolhida como a base da movimentação da impressora tridimensional e visualização de modelos virtuais;

- **Programa *Slic3r* – requisitos de especificação de *software*:** o oitavo tópico apresenta a ferramenta digital *Slic3r* e os motivos de sua escolha. O programa é de

código aberto e permite que os usuários utilizem a ferramenta como um programa isolado ou como um complemento em outras ferramentas. O *Slic3r* apresenta grande compatibilidade com o *Pronterface/Printrun*; e

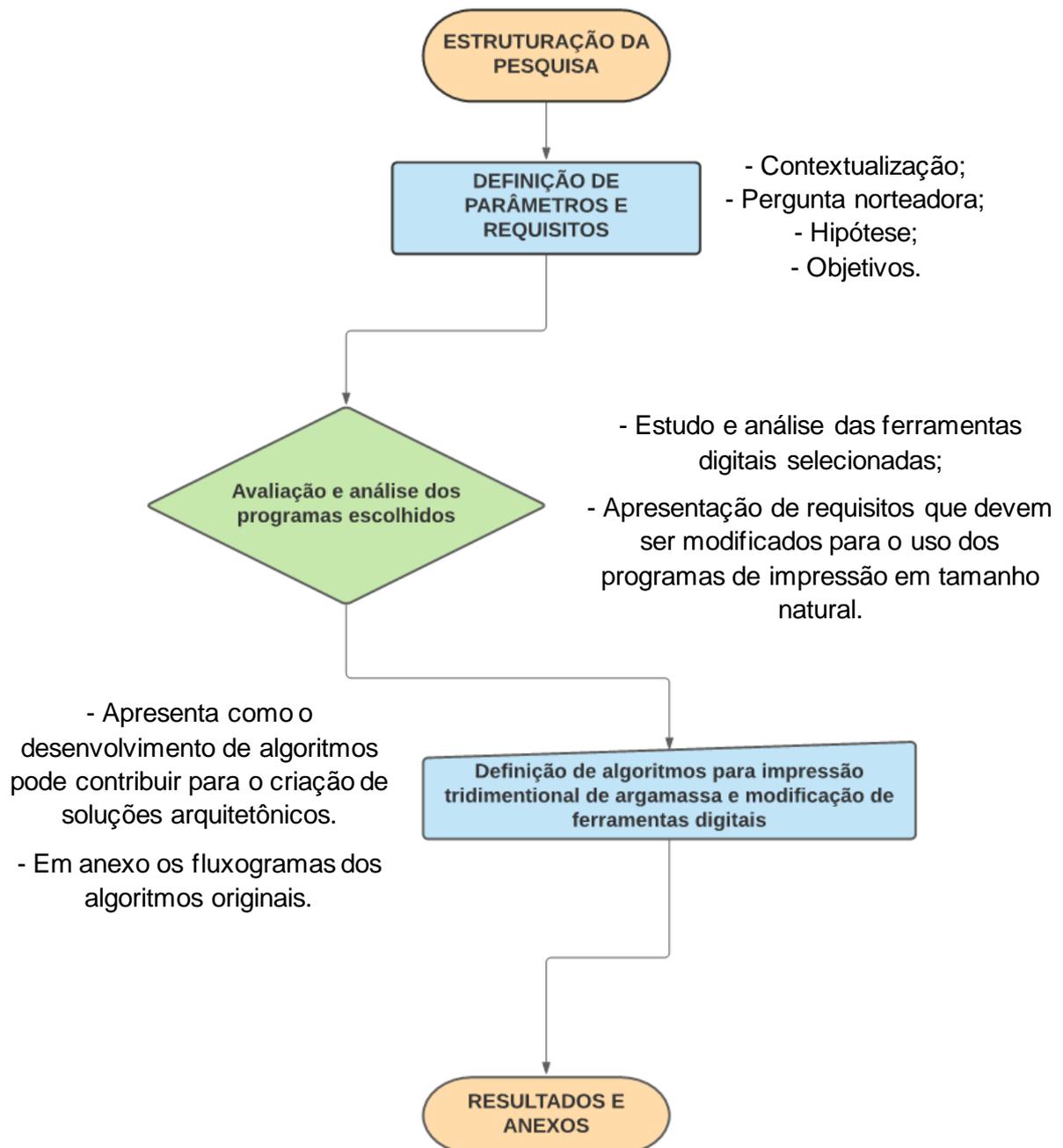
- **Resultados:** o tópico final apresenta como o desenvolvimento dos algoritmos pode contribuir para a criação de outras ferramentas ou mesmo ajudar usuários com pouca ou nenhuma experiência a utilizar impressoras tridimensionais para impressão em tamanho natural.

Ao final do trabalho é apresentado o anexo com os fluxogramas referentes aos algoritmos desenvolvidos para o estudo. Eles foram criados para facilitar o entendimento dos códigos apresentados em Português. Por fim, o anexo traz a programação desenvolvida para a impressora 3D da FAU/UnB e as linhas de códigos modificadas.

Boa leitura!

FLUXOGRAMA DE DESENVOLVIMENTO DO TRABALHO

O fluxograma, apresentado abaixo, permite o entendimento da estrutura de desenvolvimento da pesquisa e apresenta o mesmo sistema de organização de fluxograma de algoritmo. Foi necessário desenvolver primeiro os parâmetros e necessidades de impressão e, em seguida, foram analisados os programas livres. O resultado da pesquisa são os algoritmos criados para a tese.



Introdução

As novas tecnologias modificam o processo de criação, desenvolvimento e a fabricação de produtos e edificações. Impressoras tridimensionais (3D) de baixo custo são possíveis devido à queda das patentes e à liberação de projetos abertos (EHANN; WARNIER, 2014). Os projetos de impressoras liberados por pesquisadores permitiram o aperfeiçoamento e o desenvolvimento de novas propostas criadas para o processo de construção digital. Veja-se exemplo constante na Figura 1:

Figura 1 – Impressora tridimensional projetada e construída na FAU/UnB. Grupo de pesquisa do laboratório LFDC



Fonte: Balzani e Furtado, 2017.

As primeiras impressoras de projeto aberto foram apresentadas no ano de 2005 pelo professor Adrian Bowyer, da Universidade Britânica de Bath (EHANN; WARNIER, 2014). O projeto foi disponibilizado para a comunidade permitindo o desenvolvimento de impressoras que utilizam diferentes materiais tais como plásticos, sal, vidro, chocolate, argamassa de cimento e terra etc.

As impressoras de argamassa para construção se encontram em estágio inicial. O levantamento feito para a pesquisa identificou poucas iniciativas e nenhum *software* de gerenciamento para impressão de edificações em tamanho natural, escala 1:1, disponível para a comunidade acadêmica ou para comercialização. O desenvolvimento da pesquisa apresentou, por meio de testes, que *softwares* de

impressoras tridimensionais do tipo *desktop* (impressoras de mesa) não funcionam adequadamente para a definição e visualização de impressão de edificações em escala natural.

As ferramentas digitais disponíveis não permitem uma visualização adequada dos modelos pois trabalham com medidas em milímetros e os *softwares* travam ao calcular o caminho de impressão para os modelos desenvolvidos para projetos de arquitetura.

Os planejamentos de impressoras de edificações em tamanho natural demonstram potencial para a construção de habitações de interesse social e edificações de baixo custo. Todavia, é necessário ampliar a quantidade de pesquisas e propostas de projetos para elementos arquitetônicos e edificações impressas. Os pesquisadores brasileiros encontram dificuldade para utilizar máquinas de impressão tridimensional de edificações em tamanho natural e não possuem definições para a criação de ferramentas digitais de código aberto.

A utilização dessas propostas, tanto de projetos de máquinas quanto de algoritmos de computador, apresentam uma possível solução para a lacuna encontrada. O código aberto facilita o acesso às ferramentas digitais e a possibilidade de modificar máquinas para efetuar testes, produção de corpos de prova e posterior execução das edificações.

Os *softwares* de código aberto contribuíram no desenvolvimento de processos de construção digital, pois aumentaram o acesso à tecnologia e o desenvolvimento de melhorias pela comunidade de desenvolvedores (BEESLEY; CRESS, 2019). A possibilidade de criar tecnologia de código ou projeto aberto (*open-source*) propicia o surgimento de um ecossistema de inovação e disseminação de conhecimento ágil e democrático.

A digitalização do projeto de arquitetura nas últimas três décadas apresentou novas ferramentas para o desenvolvimento de projeto e representação. Os arquitetos e pesquisadores vêm se apropriando da cultura do código aberto no desenvolvimento de novas propostas arquitetônicas e construtivas. O início do movimento *open-source* pode ser atribuído ao trabalho de Linus Torvalds no desenvolvimento do sistema operacional Linux, disponibilizado gratuitamente em 1991. A criação de novas propostas *open-source* originou o sistema de licenças GNU GPL que permite o

desenvolvimento de novas versões dos códigos, mantendo o sistema aberto com suas melhorias (BEESLEY; CRESS, 2019).

Os projetos de impressoras 3D e programas de código aberto gratuitos contribuem para que arquitetos, engenheiros, construtores e pesquisadores desenvolvam propostas com uma tecnologia inserida nas previsões da Indústria 4.0, movimento para modernização dos meios de produção, construção e comunicação.

O desenvolvimento de algoritmos e requisitos para definição de ferramentas de código e projeto abertos para a impressão de argamassa permite preencher uma lacuna do conhecimento e definir como os parâmetros de impressão de materiais pastosos devem ser incorporados às tecnologias existentes.

Contudo, não existem programas de código aberto para impressoras de edificações em tamanho natural (escala 1:1) com extrusão de argamassas, o que restringe e dificulta o avanço da ciência, bem como do desenvolvimento e aplicações de processos construtivos de baixo custo com impressão 3D.

1 Contextualização

Esta seção apresenta a definição da pergunta norteadora, objetivos e métodos de trabalho.

1.1 Questão norteadora

Há uma lacuna no conhecimento comum a ser preenchida: a falta de algoritmos e requisitos para definição de ferramentas de código aberto e projetos abertos para a impressão tridimensional de edificações e artefatos arquitetônicos. Portanto, pergunta-se: Como criar algoritmos e quais são os principais parâmetros para definição de extrusão de materiais pastosos para impressoras 3D?

1.2 Hipótese

Acredita-se, nesta tese, que a lacuna no conhecimento aqui caracterizada pode ser preenchida por meio do desenvolvimento e disponibilização de algoritmos e requisitos para definição de ferramentas de código aberto para impressão 3D em tamanho natural, baseado na aplicação de parâmetros de materiais pastosos, além de requisitos para construção de edificações e componentes arquitetônicos para o processo construtivo de extrusão de argamassa em camadas.

1.3 Objetivo

O objetivo principal é o desenvolvimento e apresentação de algoritmos e requisitos de código aberto para impressão 3D em escala 1:1 baseado no processo construtivo de extrusão de argamassa.

1.4 Objetivos específicos

Os objetivos específicos são:

1.4.1 Desenvolver blocos de algoritmos

- Definir quais etapas de uso dos *softwares* de impressão 3D tipo *desktop* serão necessárias para apresentação de blocos de algoritmos e requisitos para modificação de ferramentas digitais de impressão 3D de código aberto.

1.4.2 Parâmetros das argamassas

- Desenvolver um algoritmo relacionado ao abatimento de argamassa para compensar deformações das camadas iniciais;
- Criar um algoritmo relacionado à fluidez e plasticidade das argamassas;
- Criar algoritmo para definir primeira camada de impressão;
- Desenvolver algoritmo de definição para abertura do bico de extrusão;
- Escrever algoritmo e requisitos para pausa da impressão para processo de cura;
- Definir algoritmo e requisitos para velocidade de movimentação dos eixos sem extrusão;
- Definir algoritmos e requisitos para velocidades de movimentação dos eixos no momento da impressão das camadas;
- Escrever algoritmo para preenchimento em zigue-zague; e
- Escrever algoritmo para preenchimento em grade.

O foco do estudo apresentado na tese não é a definição de traços para impressão 3D. O objetivo é definir algoritmos e requisitos para futuras ferramentas de código aberto para impressão 3D de argamassas.

Resultado esperado: apresentar soluções que contribuam para as impressões e evitem que o material derrame pelo bico de extrusão ou endureça antes de ser extrudado; e definir requisitos para auxiliar nas modificações de ferramentas de impressão de código aberto para facilitar o uso por arquitetos e profissionais da construção civil.

1.4.3 Desenvolvimento de máquina

- Colaborar no desenvolvimento de projeto e construção de uma impressora 3D para testes em centímetros;
- Definir algoritmo para base de impressão em metros;
- Descrever algoritmo para movimento no eixo Z;
- Descrever algoritmo para movimento nos eixos X e Y; e
- Apresentar a programação da impressora e exemplos de Código G.

Resultado esperado: definir os requisitos para a movimentação desejado para as impressoras de materiais pastosos em tamanho natural.

1.5 Procedimentos metodológicos

Propõe-se uma pesquisa aplicada que objetiva gerar conhecimento novo útil para o avanço da ciência e com grande potencial de aplicação prática.

Um *software* não é apenas uma ferramenta em funcionamento no computador mas também toda a documentação, configuração, parâmetros e definições especificados para o funcionamento esperado (SOMMERVILLE, 1996).

Existem diversas estratégias para o desenvolvimento de programas. O estudo se baseia no sistema *reuse-oriented software* (programa orientado para reutilização), amplamente utilizado por desenvolvedores de ferramentas digitais.

Foram escolhidos *softwares* de código aberto para observar seu uso e sua linguagem de programação. De acordo com Sommerville (1996), os parâmetros e requisitos são desenvolvidos utilizando como base ferramentas existentes, apresentando modificações e acrescentando características necessárias para o funcionamento esperado do sistema.

2 Revisão bibliográfica

A revisão bibliográfica apresenta construções de terra e a evolução da tecnologia da impressão 3D, além de como a nova fase da revolução industrial vai impactar a construção de edificações.

2.1 Construções de terra

A terra é um dos materiais mais antigos utilizados pela humanidade para a construção de abrigos e utensílios. A humanidade desenvolveu a agricultura e com ela a necessidade de se assentar. A terra foi o material base para a produção de alimentos e a matéria-prima para a construção de casas (PONTE, 2012).

As primeiras civilizações que utilizaram a própria terra local como material de construção estavam localizadas no Oriente Médio. Cidades como Jericó, aproximadamente 8.000 anos antes de Cristo, foram construídas com adobe. Outros exemplos como o Zigurate na Mesopotâmia, construído há 6.000 anos, e o Templo de Horyuji, no Japão, de 1.300 anos, demonstram como a terra é material utilizado em diferentes civilizações e culturas. A construção com terra foi amplamente utilizada pelos portugueses no processo de colonização do Brasil (TORGAL; EIRES *apud* PONTE, 2012).

A terra é um material ecológico, de baixo custo, resistente ao fogo e com grande inércia térmica. O fato de ser um material natural, abundante e encontrado no próprio local da obra faz com que construções de terra sejam uma opção pertinente na busca de soluções construtivas e para solucionar problemas como o déficit habitacional. Além disso, é o material com menor impacto ambiental existente (MACEDO; CHADIWALA *apud* PONTE, 2012).

As técnicas de construção com terra são versáteis e foram aplicadas largamente na arquitetura colonial brasileira. A terra é utilizada em técnicas milenares e atualmente percebe-se o uso do material no desenvolvimento de tecnologias inovadoras (FARIA; NEVES, 2011).

No Brasil, os mecanismos de construção com terra foram trazidas pelos portugueses e africanos. Diversas técnicas foram empregadas tais como o pau a

pique, adobe, taipa de pilão, entre outras. No século XX, foi iniciado o estudo para construção de edificações com solos estabilizados por cimentação. O solo cimento, por exemplo, é empregado para a fabricação de blocos comprimidos e tijolos. O solo é adensado em moldes por compactação ou prensagem e pode ter suas características físico-mecânicas melhoradas com o processo de estabilização granulométrica, a qual consiste na mistura de proporções diferentes de terras. A estabilização química é feita pela soma de aditivos químicos à terra, geralmente aglomerantes como o cimento ou cal (NEVES; MILANI, 2011).

É interessante comparar o sistema construtivo superadobe com a impressão 3D. A técnica foi criada pelo arquiteto iraniano Nader Khalili, em 1995. Consiste na sobreposição de camadas de sacos de polipropileno, ou outros materiais, em formato de tubos preenchidos com terra. O sistema é compactado camada a camada, conferindo resistência física e mecânica. O superadobe permite a construção de edificações em formatos de domo, arcos e conchas (ASHOK; SURESH; VENKATESWARULU, 2016). A estratégia ganhou popularidade por ser uma construção que gera pouco impacto ambiental. A terra utilizada pode ser corrigida com a adição de estabilizantes como cimento, contribuindo para a resistência do material e para manter a forma desejada (Ibidem, 2016).

O superadobe apresenta vantagens para a construção de casas de interesse social devido à facilidade por ter a terra como elemento construtivo básico, além de apresentar a qualidade de ser uma construção rápida. As paredes de terra são isolantes térmicos com baixo custo, baixa manutenção e permite edificações autoportantes (BEIRÃO; SANTOR, 2016). A impressão de argamassa de terra e cimento apresenta um produto com qualidades similares ao superadobe e com a vantagem de ser mais rápida, precisa e reduzir a mão de obra necessária. A seguir, pode ser vista uma edificação utilizando superadobe, conforme Figura 2:

Figura 2 – Edificação de superadobe



Fonte: Nunes, 2021.

A interseção entre a tecnologia de impressão 3D com argamassas de terra estabilizada e cimento como insumos permite o desenvolvimento de propostas de baixo custo e que podem ser apresentadas como soluções para habitações de interesse social. A tecnologia facilita a personalização das habitações de interesse social sem aumentar o custo ou tempo de obra (CARVALHO; DEIRÓ, 2018).

2.2 Evolução da fabricação

A Revolução Industrial é dividida em fases definidas pelos sistemas de fabricação e energia utilizados. A Primeira Revolução Industrial, iniciada em 1760, foi referente à mecanização dos processos de fabricação e ao surgimento das máquinas e transportes a vapor. Produtos e materiais de construção puderam ser transportados em grande quantidade por locomotivas e fábricas aumentaram a velocidade e capacidade de produção da Inglaterra. A fábrica se transformou em uma nova tipologia arquitetônica necessária no contexto urbano e novos materiais passaram a ser utilizados em grande quantidade na construção, tais como o ferro e o vidro (DAVID; KIM; XU, 2018).

A Segunda Revolução Industrial foi marcada por um processo de aprimoramento das técnicas de fabricação e pela utilização da eletricidade, pelo uso de materiais mais resistentes como o aço e pelo surgimento da indústria química. Esse período foi marcado pela expansão da industrialização em outros países e pela produção em massa e automatização dos trabalhos (ALBINO; FONTANELA; SANTOS, 2020).

O fim da Segunda Guerra Mundial marcou o início da Terceira Revolução Industrial, que, pelo caráter técnico-científica da segunda metade do século XX, permitiu o surgimento de avanços como a informática, eletrônica, internet e computadores pessoais (Ibidem, 2020).

O avanço na tecnologia computacional e nos meios de fabricação e produção nas primeiras décadas do século XXI define a transição para uma nova fase industrial, a quarta revolução ou Indústria 4.0.

A Indústria 4.0, ou, ainda, manufatura avançada, é caracterizada pela revolução digital e tecnológica, ou seja, a integração e o controle por meio de sensores e equipamentos conectados à internet (Ibidem, 2020). É o primeiro movimento tecnológico premeditado e não observado posteriormente. Apresenta grandes oportunidades para empresas e países com a melhoria da produtividade e dos sistemas operacionais (FUKUYAMA, 2018).

O aperfeiçoamento do processo de fabricação apresenta ganhos para a qualidade de vida. Contudo, também apresenta problemas que devem ser levados em consideração, tais como a diminuição de empregos em fábricas e como os recursos naturais serão utilizados para diminuir o impacto causado no planeta (FUKUYAMA, 2018).

A construção civil se insere no contexto da Quarta Revolução Industrial, com o advento de tecnologias de construção digital, com máquinas de corte computadorizadas, braços robóticos utilizados para construção de edificações e impressoras 3D. Diversos centros de pesquisa estão desenvolvendo técnicas para aprimorar a construção com impressoras 3D de grande formato. A impressão permite a fabricação de elementos arquitetônicos com formas complexas e a construção de edificações, diminuindo o tempo de obra e reduzindo o número de trabalhadores necessários.

2.3 A Quarta Revolução Industrial

A produção industrial como a conhecemos está sofrendo mudanças radicais em seus processos de fabricação. A maneira como as indústrias e países consomem energia, alimentos, roupas, aparelhos eletrônicos etc. está passando por um processo

de mudança e renovação. As tecnologias de fabricação digital permitem avanços nos meios de produção e fabricação de bens e produtos.

Indústrias de diversos setores e países como Alemanha, Japão e Estados Unidos da América (EUA) têm se preocupado com a grande quantidade de lixo e refugo industrial produzido e com a possibilidade de reuso de materiais que podem ser reciclados. Essas potências tecnológicas estão buscando meios para criar um ciclo produtivo que consiga integrar os processos de extração de recursos, produção e descarte de forma a otimizar o uso de matéria prima e energia. Segundo a Confederação Nacional da Indústria – CNI (2018, p. 11), “as grandes empresas industriais priorizam tecnologias digitais para aumentar a eficiência do processo de produção e melhorar a gestão dos negócios”.

Para Schwab (2016, p. 21), a escala e o escopo das mudanças explicam por que as rupturas e as inovações atuais são tão significativas. A velocidade da inovação em termos de desenvolvimento e ruptura está mais rápida do que nunca.

As tecnologias de produção, comunicação e transporte que demoravam décadas para se tornarem economicamente viáveis, acessíveis ao público e para possuir infraestrutura para sua distribuição, estão chegando mais rapidamente à população e modificando os padrões de comportamento.

É nesse contexto que a Quarta Revolução Industrial ganha repercussão, anunciando uma nova onda de mudanças nos modos de produção e consumo. O processo de renovação das tecnologias disponibiliza novos meios para o desenvolvimento de ideias, conceitos e para sua materialização de forma mais rápida, precisa e barata. A distribuição de projetos, códigos e a colaboração remota, em tempo real ou não, é garantida por meio da internet. A sociedade conectada observa a criação de novas tecnologias, serviços e utilidades inéditas.

2.4 A Indústria 4.0

Em 2011, representantes dos setores de negócios, políticos e acadêmicos alemães promoveram a iniciativa intitulada Indústria 4.0, como forma de reforçar a competitividade da indústria de manufatura da Alemanha.

O governo alemão anunciou que a Indústria 4.0 se tornaria parte do chamado “High-Tech Strategy 2020 for Germany¹”. A partir daí, um grupo de especialistas chamado “Industrie 4.0 Working Group” lançou, em 2013, as primeiras recomendações para sua implementação.

O termo Indústria 4.0 tem sido comumente associado à Quarta Revolução Industrial por compartilharem ideias similares e por conta do pioneirismo alemão. Para Santos et al. (2018, p.111), “a Indústria 4.0 é um processo que abrange um conjunto de tecnologias de ponta ligadas à internet com objetivo de tornar os sistemas de produção mais flexíveis e colaborativos”.

O nível de complexidade, bem como as tecnologias empregadas para que esse novo processo de produção seja colocado em prática, são alguns dos pontos que apresentam o início de uma nova revolução dos processos de fabricação e consumo.

É a primeira vez que uma revolução de processos de produção é anunciada, e não apenas registrada posteriormente como nas três últimas revoluções industriais, marcadas pelo uso de novas tecnologias de transporte e de novas matrizes energéticas como o vapor, petróleo e as evoluções nos sistemas de transporte como as ferrovias e os transatlânticos. Os avanços nas tecnologias de comunicação também causaram grande impacto, partindo do telégrafo no século XIX até a evolução da computação e da internet no pós-Segunda Guerra Mundial, em meados do século XX.

Os países mais avançados tecnologicamente estão desenvolvendo estratégias para planejar a Quarta Revolução Industrial, objetivando que a inserção desses novos processos de produção, serviços e comunicação seja feita de forma integrada e interconectada.

Para que esse objetivo possa ser alcançado, é necessário um alto nível de comunicação entre as diversas tecnologias que formam a cadeia da Indústria 4.0, conforme Figura 3 a seguir:

1 Em tradução livre, Estratégia de Alta Tecnologia para a Alemanha em 2020.

Figura 3 – Esquema ilustrativo dos 9 pilares da Indústria 4.0



Fonte: www.pollux.com.br, 2016.

Segundo BCG (2016), podemos descrever esses nove pilares da seguinte maneira:

Big Data e Analytics: no contexto da Indústria 4.0, a coleta e avaliação abrangente de grandes volumes de dados provenientes de muitas fontes diferentes em altíssima velocidade, equipamentos e sistemas de produção, bem como sistemas de gerenciamento corporativo e de clientes, serão o padrão para apoiar a tomada de decisões em tempo real.

Robôs autônomos: os robôs irão eventualmente interagir uns com os outros e com humanos. Os avanços nos sistemas de aprendizagem de máquina e o barateamento dos custos permitirão uma variedade maior de recursos e possibilidades de fabricação com maior precisão e em lugares inóspitos ou em situações perigosas aos seres humanos.

Simulação: as simulações serão usadas mais extensivamente em operações de fábricas para aproveitar dados em tempo real e espelhar o mundo físico em um modelo virtual. Isso permitirá que os operadores testem e otimizem as configurações de máquinas para o próximo produto da linha de produção no mundo virtual antes da mudança física, reduzindo assim os tempos de configuração das máquinas e aumentando a qualidade.

Realidade aumentada: os sistemas baseados em realidade aumentada oferecem suporte a uma variedade de serviços, como a seleção de peças em um depósito e o envio de instruções de reparo por meio de dispositivos móveis. As empresas farão um uso muito mais amplo da realidade aumentada para fornecer aos trabalhadores informações em tempo real para melhorar a tomada de decisões e os procedimentos de trabalho. Um exemplo é o aparelho “Oculus” da empresa norte-americana Facebook. O “Oculus” é um conjunto com visor e controles que permitem fazer projetos em 3D, visualizar uma maquete virtual por dentro ou modificar os acabamentos e móveis de um ambiente [cf. Figura 4].

Integração de sistemas: é a centralização de sistemas e dados que circulam de maneira mais rápida e eficiente entre os setores de uma empresa ou organização. A união de distintos sistemas e *softwares* para gestão de estoque, pagamentos, atendimento ao cliente entre outros. Empresas, departamentos, funções e recursos funcionam de forma mais coesa, conforme as redes de integração de dados entre empresas evoluem e permitem a criação de cadeias de informação automatizadas.

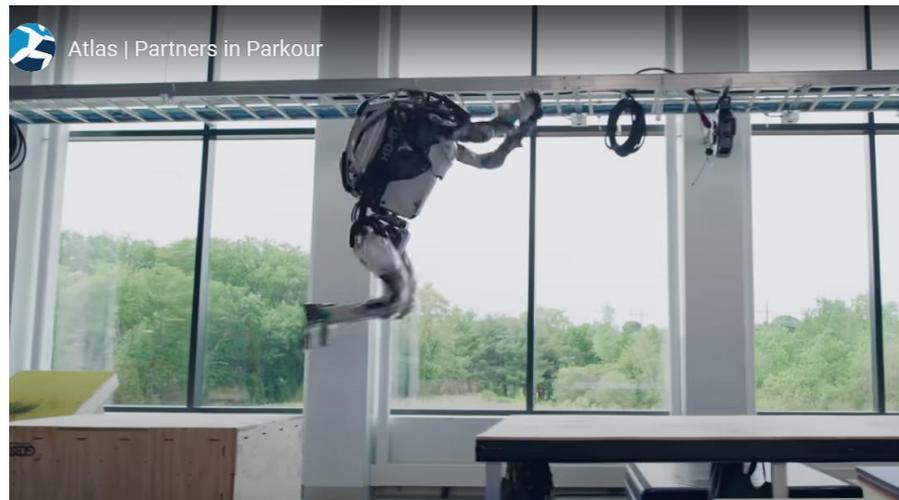
Cyber segurança: o avanço da conectividade e do uso de *softwares* para comunicação por meios de mensagem ou ligações feitas pela internet, acesso à conta de bancos de forma remota e sistema de defesa que acompanham a Indústria 4.0, aumentou a necessidade de proteger sistemas industriais críticos e linhas de manufatura contra ameaças à segurança cibernética. Como resultado, os países e empresas de tecnologia estão investindo em melhorias para os sistemas de comunicação e para a segurança dos seus sistemas de controle digitais conectados à internet, bem como gerenciamento sofisticado de identidade e acesso de máquinas e usuários.

Nuvem: o sistema de gerenciamento de informações e armazenamento em nuvem permite que as empresas ganhem produtividade e precisão. O armazenamento em nuvem é feito por meio de grandes centros de *backup* em que as empresas e usuários podem acessar seus documentos de qualquer dispositivo conectado à internet. Outro aspecto relevante é a computação ou processamento em nuvem que permite utilizar computadores potentes de forma remota para produzir o processamento necessário para um trabalho.

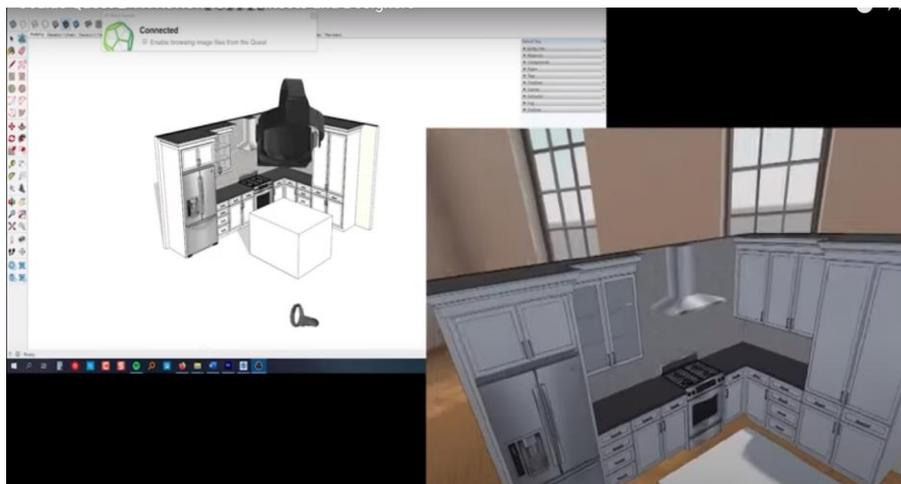
Internet industrial: os microcomputadores e a conectividade com internet de alta velocidade permite que eletrodomésticos, objetos e sistemas industriais sejam enriquecidos com computação integrada. A internet industrial permite que os ambientes e os equipamentos se comuniquem e interajam entre si. Esse processo também descentraliza análises e tomadas de decisão, permitindo respostas em tempo real [cf. Figura 6].

Manufatura aditiva: a manufatura aditiva é uma das tecnologias que permitem uma grande variedade de possibilidades. Diversos campos de pesquisa estão utilizando a tecnologia de fabricação aditiva para desenvolver trabalhos nas áreas de desenho de produto, construção civil e na medicina. Os métodos de manufatura aditiva são amplamente usados para produzir pequenos lotes de produtos personalizados que oferecem vantagens de construção, como projetos complexos para resolver problemas individuais sem aumento do custo de fabricação.

A empresa estadunidense Boston Dynamics desenvolve robôs de alta tecnologia, capazes de correr, pular, pegar objetos e adaptar seus movimentos em situações adversas e em terrenos acidentados, do modo como é demonstrado na Figura 4. Outros exemplos, referentes às descrições supracitadas, também podem ser visualizados nas Figuras 5 e 6:

Figura 4 – Robô Atlas

Fonte: Boston Dynamics, 2021.

Figura 5 – Oculus. Realidade Aumentada

Fonte: Facebook, 2021.

Figura 6 – Armazém de distribuição da Amazon

Fonte: Supply Chain Magazine, 2021.

2.5 O plano estratégico da manufatura avançada nos EUA

Os EUA possuem um dos maiores e mais avançados parques industriais do mundo. Como forma de manter o país competitivo, o setor industrial de alta tecnologia apresentou um diagnóstico para o governo estadunidense e, em 2012, lançou o chamado “A National Strategic Plan For Advanced Manufacturing²”, documento similar ao lançado pela Alemanha, com estratégias e objetivos para o novo estágio da produção industrial.

A manufatura avançada é descrita, segundo IEDI (2019, p. 268), como “uma família de atividades que dependem do uso e coordenação de informações, automação, computação, *software*, detecção em rede e/ou utilizam materiais de ponta”. O movimento direciona as habilidades emergentes das ciências físicas e biológicas, como a nanotecnologia, a química e a biologia. Envolve tanto novas formas de fabricação de produtos existentes quanto a fabricação de novos produtos que surgem das novas tecnologias avançadas.

Em 2016, foi apresentado o “Advanced Manufacturing: a snapshot of priority technology areas across the Federal Government³”. O documento apresenta informações sobre tecnologias emergentes ou consolidadas mais relevantes para o futuro da indústria estadunidense, tanto para a competitividade do setor industrial quanto para a segurança nacional, de forma a manter a posição de destaque dos EUA em relação às outras nações consideradas desenvolvidas (IEDI, 2019).

2.6 O Japão diante da Indústria 4.0

O Japão é conhecido por sua capacidade organizacional e por seu desenvolvimento tecnológico. É um dos países pioneiros no esforço para implementar políticas internas de desenvolvimento referentes às tecnologias ligadas à Indústria 4.0.

O país desenvolveu uma estratégia de revitalização. O governo incluiu um conjunto de medidas proativas voltadas à sua aceleração no contexto da Quarta

2 Em tradução livre, “Um Plano Estratégico Nacional para Manufatura”.

3 “Manufatura Avançada: um panorama das áreas de tecnologia prioritária em todo o Governo Federal”, em livre tradução.

Revolução Industrial, direcionada para o desenvolvimento de tecnologias como a Internet das Coisas (IOT), *Big Data*, robótica avançada e inteligência artificial, para fortalecer a competitividade da economia japonesa (IEDI, 2019).

A política idealizada de revitalização visa também a criação de novos mercados, em setores da indústria, da medicina, da agricultura e no setor de distribuição e logística.

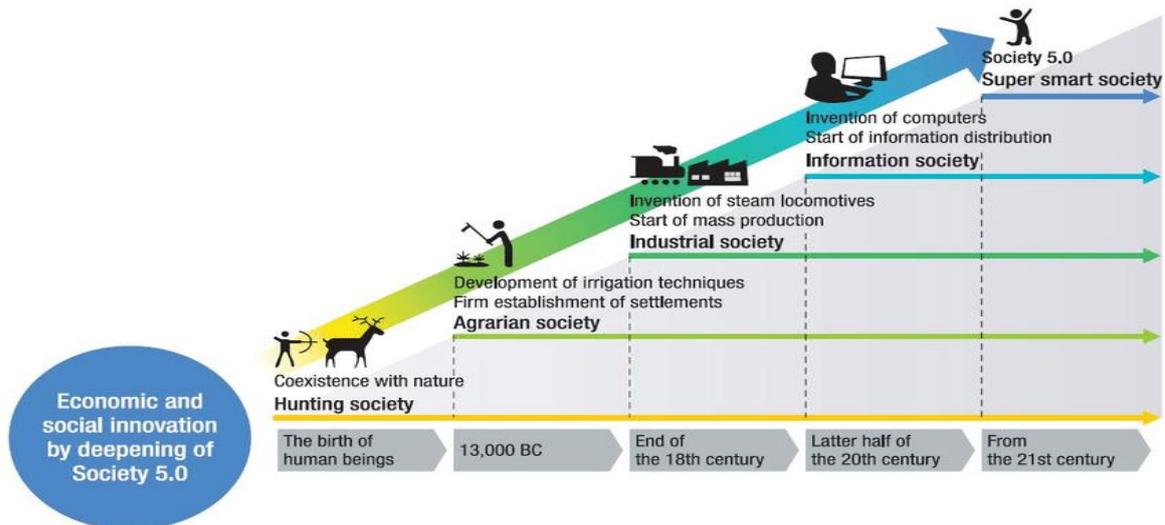
Foi a partir dessa nova estratégia que criou-se a “Robotic Revolution Initiative” (RRI) (Iniciativa de Revolução Robótica), cujo objetivo é colocar o Japão como líder mundial por meio do desenvolvimento e do uso de robôs integrados com sistemas de Internet das Coisas.

O governo japonês criou em 2017 o conceito de “Connected Industries” (CI) (Indústrias Conectadas), que tem como foco gerar uma cadeia de indústrias totalmente conectadas com o uso de tecnologias como a Internet das Coisas, inteligência artificial e *Big Data*. O Japão espera alcançar por meio desse projeto a Sociedade 5.0.

A Sociedade 5.0 é um modelo de organização social em que as tecnologias relacionadas à Quarta Revolução Industrial serão usadas para criar soluções com foco nas necessidades humanas. O objetivo é equilibrar questões como o avanço econômico e a resolução de problemas sociais.

O conceito relacionado à Sociedade 5.0 é alcançar um novo patamar para a sociedade, tomando por base as outras quatro fases já vivenciadas, conforme descrição da Figura 7 a seguir:

Figura 7 – Evolução das sociedades em 5 fases



Source: Prepared by the author based on material from the Japan Business Federation (Keidanren) "Japan's initiatives — Society 5.0"; Y. Harayama, "Society 5.0: Aiming for a New Human-centered Society", Hitachi Review, vol. 66, no. 6, 2017, pp. 556–557

Fonte: Fukuyama, 2018.

- Sociedade da caça (1.0): Época em que os seres humanos viviam como nômades, caçadores-coletores;
- Sociedade da agricultura (2.0): Momento em que o ser humano desenvolveu técnicas de cultivo. A agricultura permitiu ao ser humano se fixar em um local e desencadeou o surgimento dos primeiros assentamentos, povoados humanos e, conseqüentemente, das primeiras cidades;
- Sociedade industrial (3.0): É a sociedade da Primeira Revolução Industrial. Surgimento dos primeiros motores a vapor, máquinas automatizadas e o rápido crescimento urbano. A Primeira Revolução Industrial definiu mudanças nos sistemas de produção e nos hábitos de consumo. A sociedade industrial iniciou problemas relacionados à habitação, saneamento básico nas cidades e aceleração do processo de poluição do meio ambiente;
- Sociedade da informação (4.0): Sociedade pós-Segunda Guerra Mundial, Terceira Revolução Industrial, saindo dos processos analógicos e migrando para uma sociedade digital. Apresentou avanços nos sistemas de comunicação como o uso de satélites e automação das indústrias;

O Japão acredita que na Sociedade 5.0 haverá uma hiperconexão entre os computadores e os seres humanos, resultando em um modo de vida mais inteligente, eficiente e sustentável. O objetivo dos idealizadores da Sociedade 5.0 é criar uma

sociedade centrada no ser humano em que tanto o desenvolvimento econômico individual quanto a resolução de problemas sociais sejam uma realidade, e as pessoas possam usufruir de uma boa qualidade de vida (FUKUYAMA, 2018).

2.7 China e o plano estratégico *Made In China 2025*

A China é outra potência asiática que se posicionou para se adaptar às mudanças tecnológicas relacionadas à Indústria 4.0. O país possui um sistema industrial amplo e competitivo e busca ampliar a eficiência de seus sistemas por meio da inovação, uso eficiente dos recursos, qualidade industrial e ampliação do nível de digitalização da indústria (SANTOS, 2018).

Diante dessa perspectiva, em 2015, o governo chinês lançou o documento intitulado “Made in China 2025” com direcionamentos para implementar uma nova política industrial em seu território, transformando o país em uma potência industrial mundial até 2049, ano em que se comemorará o centenário de fundação da Nova China.

A iniciativa lista as estratégias definidas para garantir que a China avance em sua produção industrial. A proposta visa incluir princípios de inovação, economia de energia, reciclagem, otimização de estrutura e outros fatores que permeiam o setor industrial. O objetivo é transformar a China de um país de mão de obra de baixo custo para um país de engenheiros.

O documento “Made in China 2025” é dividido em três metas com datas de conclusão definidas para garantir que todo o programa seja realizado em sua totalidade até o ano de 2050, conforme Figura 8, abaixo.

- A primeira meta é transformar o país na maior potência manufatureira do mundo em 10 anos, concluindo esse feito até 2025. Nessa fase, a China quer consolidar sua industrialização e aprimorar outros pontos importantes para garantir sua evolução tecnológica, tais como a melhoria das redes digitais, redução dos níveis de poluição, aumento da capacidade de inovação e posição de destaque no cenário global entre as potências industriais;
- A segunda meta é se tornar líder mundial no quesito inovação industrial e tecnológica. O prazo para essa fase é até o ano de 2035, e o país espera

desenvolver sua capacidade produtiva para alcançar um nível intermediário entre as maiores potências globais;

- A terceira fase é consolidar o programa transformando a China em líder mundial até 2049. Um dos principais objetivos é celebrar o Centenário da Nova China apresentando-se como uma potência industrial inovadora e de alta tecnologia. Isso permitirá que o país esteja em vantagem nos setores de manufatura e fabricação digital.

Figura 8 – Objetivos das iniciativas nacionais do MIC 2025

China - Objetivos das Iniciativas Nacionais do MIC 2025		
Iniciativas	Detalhes	Objetivos
Centros de P&D e inovação	Impulsionar avanços tecnológicos e inovação em áreas-chave como tecnologia de informação e comunicação de próxima geração, manufatura inteligente, novos materiais, manufatura aditiva e produtos farmacêuticos.	Estabelecer 15 "centros nacionais de inovação industrial" até 2020; 40 até 2025
Projetos de manufatura inteligente	Principais empresas chinesas envolvidas na criação e otimização de projetos e técnicas de fabricação inteligente, digitalização de fábricas e personalização de cadeias de suprimentos.	Diminuir os custos operacionais em 30%, reduzir os tempos de produção em 30% e reduzir as taxas de defeito em 30% até 2020; e, em seguida, redução de 50% nas taxas de custos, horários e defeitos até 2025.
Bases industriais	Estabelece novos centros de pesquisa para acelerar o desenvolvimento dos principais componentes industriais, técnicas, materiais e tecnologia de produção, denominados "Quatro Básicos".	Ser autossuficiente para 40% dos componentes e materiais essenciais em setores-chave - aeroespacial; telecomunicações; produção e distribuição de energia; transportes e eletrodomésticos - até 2020; e então 70% até 2025.
Projetos industriais ecológicos	Realizar projetos em eficiência energética, proteção ambiental, uso de recursos, reutilização e tecnologias com baixas emissões de carbono.	Construir 1.000 fábricas verdes e 100 parques industriais verdes até 2020. Reduzir a emissão de poluentes primários em 20%. Alinhar o consumo de energia por unidade com níveis mundiais avançados até 2025.
Equipamentos industriais de ponta	Desenvolvimento de projetos inovadores, de ponta, focados na indústria em veículos aeroespaciais, ferroviários, de nova energia, redes marinhas, redes inteligentes, máquinas-ferramentas de ponta, equipamentos nucleares e médicos.	Realizar a P & D independente nos setores prioritários para alcançar expressivo crescimento na participação de mercado da China no IP para equipamentos de alto valor até 2025

Fonte: IEDI, 2018.

2.8 Agenda brasileira para a indústria 4.0

O Brasil traçou no ano de 2017 uma estratégia para ingressar na Indústria 4.0 e acompanhar as potências globais criando o Grupo de Trabalho para a Indústria 4.0 (GTI 4.0) responsável por desenvolver uma agenda nacional sobre o tema. O GTI 4.0 possui mais de 50 instituições representativas, entre empresas privadas, agências governamentais e sociedade civil organizada que promovem debates sobre a temática para propor ações visando a melhoria da indústria nacional.

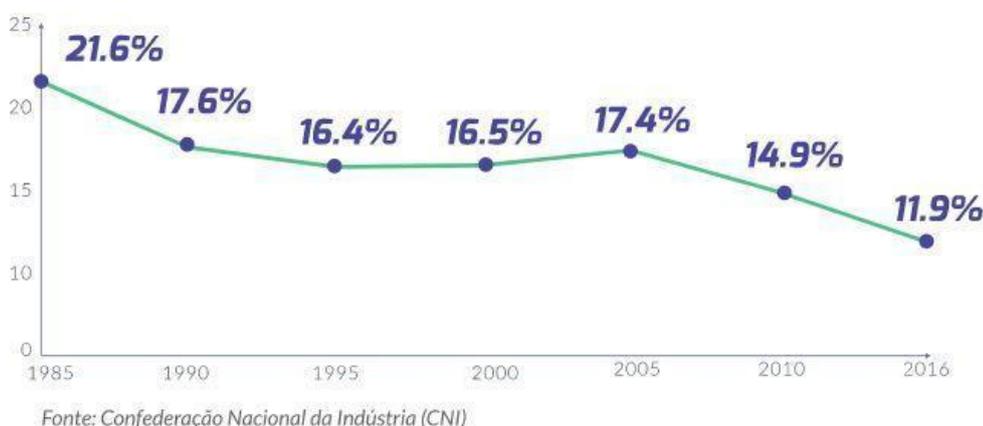
No ano de 2019, o grupo lançou a Agenda Brasileira para a Indústria 4.0. Os ministérios da Economia e da Ciência, Tecnologia, Inovações e Comunicações lançaram a Câmara Brasileira da Indústria 4.0, com o objetivo de unificar as políticas públicas do Governo Federal de fomento da manufatura avançada e da Internet das Coisas para otimizar recursos financeiros, profissionais e modernizar o parque industrial brasileiro.

A Câmara Brasileira da Indústria 4.0 acredita que as principais tecnologias que permitirão a fusão dos mundos físico, digital e biológico na indústria são a manufatura aditiva, a Inteligência Artificial, a Internet das Coisas, a Biologia Sintética e os Sistemas CiberFísicos.

No *site* apresentado pelo governo brasileiro, disponível até o ano de 2020 em www.industria40.gov.br, eram indicados alguns dos desafios que o Brasil precisa enfrentar para conseguir uma posição de destaque diante desse novo cenário industrial.

O GTI 4.0 apresenta dados que indicam que a indústria reduziu sua participação no crescimento do Produto Interno Bruto (PIB). Sua participação encolheu para aproximadamente 11% (cf. Figura 9) nas duas primeiras décadas do século XX. Em meados da década de 1980, a indústria havia atingido mais de 20% da participação no PIB. Esse fato se deve às mudanças na estrutura produtiva do país e dos novos modelos de negócios trazidos pela disrupção tecnológica.

Figura 9 – Participação do setor de transformação industrial no PIB (%) 1985 – 2016



O Brasil ocupa, atualmente, a 69ª posição no Índice Global de Inovação. Esse índice avalia critérios de *performance* de diferentes países no quesito inovação, tomando por base parâmetros como crescimento da produtividade, investimentos em pesquisa e desenvolvimento (P&D), educação e exportações de produtos de alta tecnologia. O *ranking* pode ser visto na Figura 10:

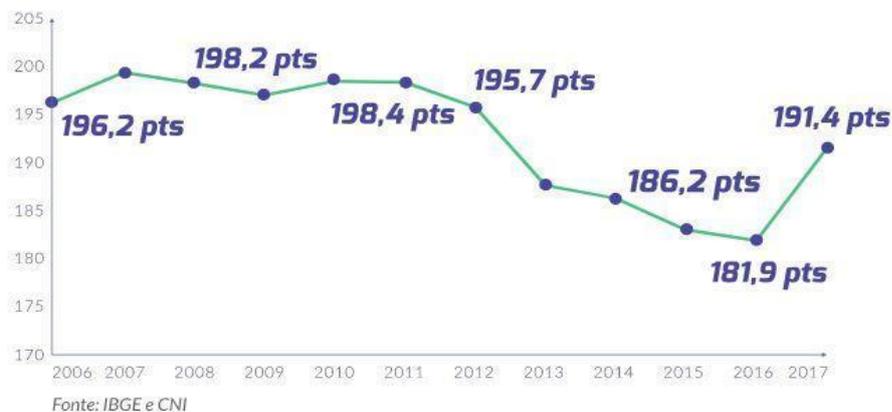
Figura 10 – Índice global de inovação: países mais inovadores.

Posição	País	Pontos
1º	Suíça	67,6 pts
2º	Suécia	63,8 pts
3º	Países Baixos	63,3 pts
4º	EUA	61,4 pts
5º	Reino Unido	60,8 pts
6º	Dinamarca	58,7 pts
7º	Cingapura	58,6 pts
8º	Finlândia	58,4 pts
69º	Brasil	33,1 pts

Fonte: Industria40, 2016.

Segundo dados informados pelo IBGE e pela CNI, a produtividade da indústria brasileira caiu mais de 7 pontos dos anos de 2006 a 2016 (*vide* Figura 11). Esse índice demonstra como o Brasil vem perdendo competitividade na comercialização de produtos industrializados e a falta de investimentos para melhoria do setor industrial e da inovação tecnológica.

Figura 11 – De 2006 a 2016, a produtividade da indústria brasileira caiu mais de 7 pontos.



Fonte: Industria40, 2016.

No *ranking* do Índice Global de Competitividade da Manufatura, o Brasil saiu da 5º posição, em 2010, para a 69º, em 2016, conforme demonstrado pela Figura 12:

Figura 12 – Índice Global de Competitividade da Manufatura.

<i>Posição</i>	<i>País</i>	<i>Pontos</i>
1º	Suíça	67,6 pts
2º	Suécia	63,8 pts
3º	Países Baixos	63,3 pts
4º	EUA	61,4 pts
5º	Reino Unido	60,8 pts
6º	Dinamarca	58,7 pts
7º	Cingapura	58,6 pts
8º	Finlândia	58,4 pts
69º	Brasil	33,1 pts

Fonte: Industria40, 2016.

Mesmo diante desse cenário, o governo brasileiro vislumbra um futuro promissor para a indústria a partir do momento em que o setor fizer a migração para os meios de produção vinculados à Indústria 4.0.

Nesse contexto de inovação e revolução tecnológica, a arquitetura e a construção civil devem se posicionar para acompanhar o surgimento de novas técnicas de construção e fabricação de edificações. A manufatura aditiva, impressão 3D, apresenta uma variedade de possibilidades para o desenvolvimento de estudos em escala reduzida e produtos com impressoras 3D do tipo *desktop*.

A rápida evolução de impressoras tridimensionais apresenta a possibilidade de imprimir com diferentes tipos de materiais tais como vidro, areia, gesso, ferro e materiais pastosos.

As impressoras em tamanho natural de materiais pastosos permitem que arquitetos desenvolvam tanto artefatos arquitetônicos quanto a construção de edificações.

É necessário que o Brasil entenda o rápido desenvolvimento da Quarta Revolução Industrial. O país deve se posicionar junto às potências tecnológicas para não permanecer defasado técnica e competitivamente no mercado internacional.

2.9 Fabricação e construção digital aditiva

A manufatura aditiva engloba um grupo de máquinas que produzem objetos a partir de modelos virtuais. As primeiras impressoras 3D viáveis para a indústria foram desenvolvidas nos anos 1980. Tecnologias como a Estereolitografia (*stereolithography*), impressão com resina sensível à luz, desenvolvida por Chuck Hull em 1984, a Sinterização Seletiva a Laser (SLA), impressão com material em pó agregado com laser, desenvolvida por Carl Deckard em 1988, e a impressão por Deposição de Material Fundido (FDM), desenvolvida por Scott Crump em 1988, permitiram que a indústria aplicasse a impressão 3D no desenvolvimento de protótipos e produtos.

A popularização das tecnologias de impressão 3D ganhou força a partir do ano de 2005, com a queda de patentes que impediam a outras empresas e pessoas a fabricação e comercialização de impressoras. O pesquisador Adrian Bowyer desenvolveu uma impressora de código e projeto aberto (*vide* Figura 13). O projeto foi pensado para que uma impressora 3D pudesse imprimir as peças de outra impressora, permitindo sua replicabilidade para a comunidade de desenvolvedores conectados à internet (EHANN; WARNIER, 2014).

O projeto Replicating Rapid Prototyper (RepRap), Prototificadora rápida de replicação, permite a desenvolvedores e pesquisadores construir impressoras 3D com peças e equipamentos encontrados no mercado, como motores de passo e controladoras de uso geral, e com peças impressas em outras impressoras similares.

Figura 13 – Impressoras 3D de baixo custo RepRap. Projeto Adrian Bower.



Fonte: Ehann e Warnier, 2014.

A popularização das impressoras 3D permitiu que pesquisadores utilizassem o projeto base para construir modelos diferentes e testar o uso de diferentes materiais. A tecnologia de impressão 3D utilizada para a construção de elementos arquitetônicos e edificações evoluiu muito, um exemplo é o projeto “Digital Grottesque”, apresentado pela Figura 14, a seguir:

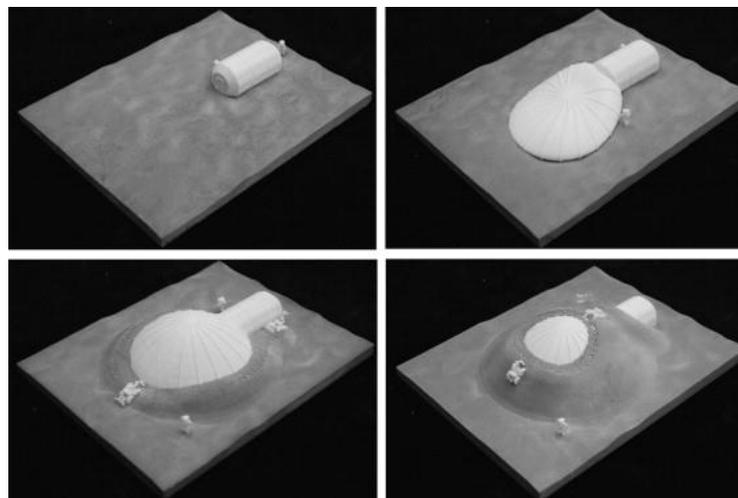
Figura 14 – Digital Grottesque, Suíça, 2013.



Fonte: Hansmeyer *apud* Krugelis, 2018.

Agências aeroespaciais, como a National Aeronautics and Space Administration (Nasa) e a Agência Espacial Europeia (ESA), estão investindo em projetos visando a possibilidade de fabricar possíveis estações lunares e marcianas com impressoras 3D. O escritório Foster + Partners, em parceria com diversas empresas e pesquisadores, ganhou, em 2009, um concurso da ESA com uma proposta de impressão 3D na lua, conforme demonstrado pela Figura 15, (KESTELIER et al., 2009):

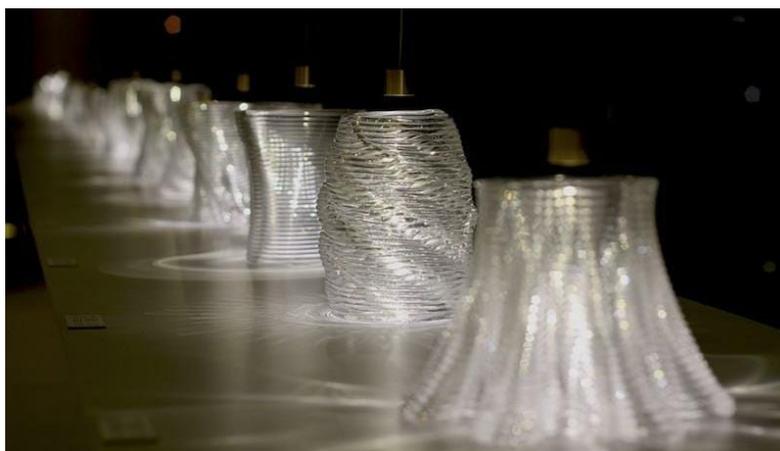
Figura 15 – Proposta para estação Lunar, Foster + Partners.



Fonte: Cesaretti et al., 2009.

Diferentes tipos de materiais estão sendo testados para a impressão tridimensional em tamanho natural ou para a impressão de elementos arquitetônicos. O *Media Lab.* do Instituto de Tecnologia de Massachusetts (MIT), sob o comando da pesquisadora Nery Oxman, desenvolveu, no ano de 2015, uma impressora capaz de extrudar vidro incandescente (cf. Figura 16) e criar formas variadas, utilizando o processo de manufatura aditiva por camadas (KLEIN et al., 2020).

Figura 16 – Projeto “Glass I”



Fonte: *Media Lab.* MIT, Nery Oxman, 2015.

A impressão tridimensional com materiais pastosos tem ganhado espaço. A possibilidade de imprimir com argila, argamassas de cimento e argamassas de terra tem motivado pesquisadores do mundo inteiro a trabalhar em propostas de fabricação de edifícios com manufatura aditiva.

A construção por manufatura aditiva de argamassa é apresentada por diversos pesquisadores, definindo novas estratégias de uso do material e da tecnologia de impressão. A liberdade de movimentação dos eixos de uma impressora 3D e as propriedades das argamassas definem as linhas guias para a criação de projetos e construções com custos reduzidos de mão de obra e de materiais (MARIJNISSEN; MARJOLEIN; VAN DER ZEE, 2017).

A impressão de argamassa de cimento apresenta diversas vantagens. Empresas estão criando argamassas específicas para a impressão tridimensional que causam menos impacto na natureza e reduzem o nível de dióxido de carbono (CO₂) emitido. Um exemplo é o da empresa Cybe Additive Industries que desenvolveu uma argamassa para impressão que produz até 32% menos CO₂. Outro exemplo é a

empresa chinesa Winsun, que utiliza material reciclado de edificações demolidas (MARIJNISSEN; MARJOLEIN; VAN DER ZEE, 2017).

Um dos grandes benefícios da impressão tridimensional de grande formato é a possibilidade de personalizar as edificações. A personalização é uma opção viável com a impressão 3D pois não é necessário o uso de formas ou elementos pré-fabricados fornecidos pela indústria (MARIJNISSEN; MARJOLEIN; VAN DER ZEE, 2017).

A impressão aditiva utiliza o mesmo sistema construtivo tanto para impressoras de mesa quanto para impressoras de tamanho natural. Os objetos são formados por perímetros e por preenchimentos definidos em *softwares* específicos que calculam o movimento da máquina e a quantidade de camadas de acordo com a espessura indicada pelo projetista (cf. Figura 17). É importante ressaltar que a possibilidade de imprimir camadas sobre camadas desonera a obra, mas limita a possibilidade de criar obras em balanço (MARIJNISSEN; MARJOLEIN; VAN DER ZEE, 2017).

Figura 17 – Camadas de impressão 3D em concreto



Fonte: Marijnissen e Van der Zee, 2017.

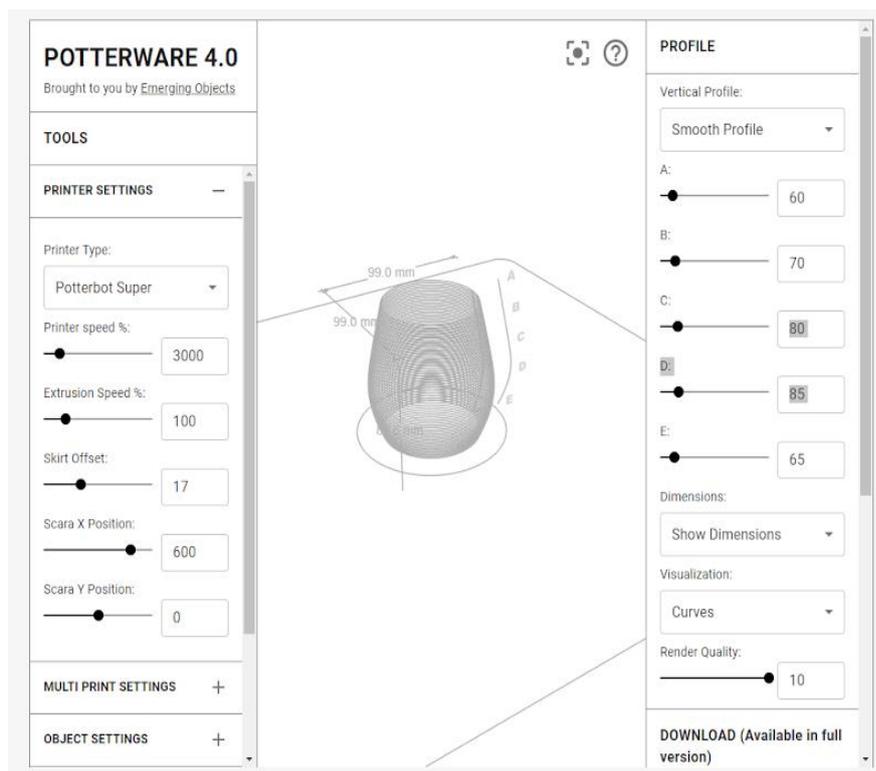
A impressão de terra permite um grande avanço no uso de materiais para impressão de edificações. A terra é um material disponível em grande quantidade, gratuito ou muito barato e de baixo impacto ambiental. A possibilidade de criar modelos acessíveis com tecnologia aberta e imprimir edificações de terra é um avanço na tecnologia (FRATELLO; RAEL, 2020).

Existem alguns exemplos de impressoras utilizadas para a construção de edificações inteiras como o projeto “Canal House Cabin”, do escritório holandês DUS

Arquitetura; o projeto “DFAB House”, da pesquisa de Gramazio e Kohler; e o pavilhão “Flotsam e Jetsam”, da empresa Branch Technologies.

É possível utilizar o *software Potterware* para impressão em argila. O programa é desenvolvido pela empresa Emerging Objects e está em sua quarta versão, como demonstrado pela Figura 18. O *software* permite o desenvolvimento de arquivos para impressão tridimensional em pequeno formato com argila. Os objetos podem variar de vasos a elementos construtivos de edificações como peças para fachadas. O programa viabiliza a utilização de diferentes tipos de impressoras e possibilita a definição de parâmetros vinculados à velocidade de movimentação dos eixos, espessura das camadas, entre outros. Contudo, não permite a utilização de parâmetros físico-mecânicos presentes das argamassas de terra ou concreto para construção civil.

Figura 18 – Interface de software Potterware



Fonte: Potterware, 2020.

A investigação de sistemas abertos para a aplicação de tecnologia de impressão 3D de argamassas é uma necessidade para que arquitetos e pessoas envolvidas na construção civil possam ter acesso à tecnologia (KONTOVOURKIS;

TRYFONOS, 2020). É necessário que trabalhos científicos investiguem a configuração de impressão com argamassa de terra e de cimento.

Existem parâmetros não abordados por estudos com outros tipos de pastas, tais como a espessura das paredes, adição de palha, fibras vegetais ou resíduos de papel *kraft* para diminuir a retração e a plasticidade do material, entre outros aspectos.

O levantamento apresentado permite alguns questionamentos relevantes referentes à impressão 3D em tamanho natural. Como podemos contribuir para ampliar o acesso à tecnologia de impressão 3D e para novas pesquisas no campo da construção digital? Como facilitar o processo de impressão de argamassas por meio de tecnologia de código aberto (máquinas e *softwares*)?

Não existem métodos ideais para a construção aditiva. As diferentes técnicas apresentam pontos positivos e negativos que devem ser considerados, tais como propriedades físicas, mecânicas e visuais exigidas para a construção da edificação projetada (ALVES; NETO, 2015).

O custo da impressão é reduzido e permite que objetos sejam produzidos de forma personalizada em pequenas quantidades. O processo de desenvolvimento de um produto ou objeto ganha rapidez, pois diminui o tempo entre projeto e produção de protótipos ou peças finais. As impressoras 3D permitem a fabricação de peças com formas orgânicas e a possibilidade de modificar os produtos de acordo com a necessidade. Outro fator de economia é a utilização apenas do material necessário para o objeto ou construção idealizado, diminuindo sobras ou perdas de material (CHAN et al., 2018).

As pesquisas com construção digital se diferenciam pelo desenvolvimento de novos materiais para impressão 3D voltados para construção civil e o desenvolvimento de impressoras de tamanho natural que produzam elementos arquitetônicos e edificações em fábricas ou no sítio escolhido para a obra.

2.10 Uso da tecnologia de fabricação e sua evolução para impressão tridimensional de edificações

Muitos protótipos são desenvolvidos em tamanho reduzido para que sejam avaliadas todas as variáveis envolvidas em um projeto de arquitetura, extrapolação de custos e de tempo, além da possibilidade de ajustes e adequação do modelo.

No entanto, é inevitável a tendência da migração para a impressão em tamanho natural, escala 1:1, embora tenha de enfrentar desafios iniciais inerentes ao processo de aprendizagem como perdas, erros nos modelos virtuais/projetos ou parâmetros de impressão. A transição do planejamento digital para a manufatura digital é uma consequência lógica. A construção digital deve fazer parte das inovações da Construção 4.0 referentes à Indústria 4.0.

Um dos aspectos mais estudados no âmbito da manufatura aditiva, e que apresenta o valor de pesquisa, é a possibilidade de desenvolvimento de diferentes tipos de impressão 3D devido ao material de deposição.

A impressão 3D inicialmente era utilizada exclusivamente para o desenvolvimento de produtos e protótipos para a indústria (KRUGELIS, 2018). Entretanto, novas oportunidades surgiram para que arquitetos e engenheiros desenvolvessem propostas com essa tecnologia. A impressão tridimensional permite uma nova forma de construir, retirando o aspecto de construção artesanal, o que, no entanto, possibilita um processo integrado que aproveita as vantagens da tecnologia 3D associadas aos padrões convencionais sem a sua completa exclusão (FRATELLO; RAEL, 2020).

A impressão apresenta benefícios como a flexibilidade arquitetônica e alta precisão (ALLOUZI; AL-AZHARI; ALLOUZI, 2020), cuja personalização não impacta no custo final da obra (MARIJNISSEN; MARJOLEIN; VAN DER ZEE, 2017); a redução do tempo de construção pela otimização do trabalho (ALEKSANDROVICH et al., 2019); a diminuição do trabalho intensivo na construção e seu alto risco aos trabalhadores, e também da poluição ambiental com o uso mais eficiente dos recursos naturais; e um maior custo-benefício.

Apesar das vantagens, as limitações e os desafios também são listados. O principal desafio apontado por vários autores é a dificuldade de introduzir reforços

estruturais no momento da impressão ou a seleção de materiais altamente resistentes, capazes de dispensar a utilização de armaduras (ALLOUZI; AL-AZHARI; ALLOUZI, 2020; ALEKSANDROVICH et al., 2019). Apesar do assunto sobre a aplicação do concreto na construção digital ter crescido rapidamente em todo o mundo, as soluções sugeridas para a incorporação de armadura ainda são rudimentares.

Outros pontos a serem considerados são a necessidade de mão de obra qualificada para executar o trabalho de operação das máquinas e acompanhamento da impressão de uma edificação e o transporte de máquinas para o local de fabricação de uma edificação ou das peças impressas em uma fábrica.

Quanto ao custo, há uma falta de conhecimento sobre a economia que essa tecnologia pode oferecer. A falta de análise de custo-benefício da impressão 3D indica uma lacuna de pesquisa a ser explorada na área de custos. É importante ressaltar que o estudo apresentado nesta tese não aborda de forma aprofundada a questão dos custos referentes a uma edificação impressa. No atual estado da tecnologia, não é possível afirmar que o custo da impressão 3D é inferior ao da construção convencional, uma vez que os parâmetros analisados não podem ser os mesmos.

Uma aplicação interessante para a tecnologia de impressão em tamanho natural com argamassa refere-se à impressão de módulos em lugares de difícil acesso com quantidade reduzida de mão de obra e em menor tempo. Tal situação se adequa às construções emergenciais, nas quais a duração e a otimização dos processos é primordial, como no caso de edificações penais ou de abrigos emergenciais.

2.11 Impressões 3D de edificações

Um dos pioneiros e principais desenvolvedores de estudos sobre essa tecnologia é o projeto “Contour Crafting”, demonstrado pela Figura 19, liderado pelo professor Behrokh Khoshnevis e baseado na Universidade do Sul da Califórnia, nos Estados Unidos da América. Florêncio et al. (2016, p. 2) explica a possível revolução da aplicação desse sistema no processo construtivo:

Aumentando a escala das impressoras 3D convencionais, esperamos construir bairros inteiros de moradias dignas, utilizando uma fração do custo e do tempo em um ambiente muito mais seguro e oferecendo uma flexibilidade formal sem precedentes.

Seus experimentos demonstraram a capacidade de imprimir uma casa inteira em vinte horas.

Figura 19 – Ilustração da impressora 3D da “Contour Crafting”



Fonte: Florêncio; Segundo e Quintella, 2016.

É interessante ressaltar que em técnicas como a “Contour Crafting”, a impressão das edificações é realizada diretamente sobre o sítio definido para a edificação, diminuindo despesas relacionadas a transporte de peças, pessoas e desonerando a obra. A empresa WinSun apresenta outra possibilidade, operando dentro de uma fábrica para produzir peças grandes, tirando proveito do ambiente controlado.

A impressão tridimensional em concreto vem sendo estudada e testada desde meados dos anos de 1990, no estado da Califórnia/EUA. Um dos métodos estudados é o sistema de manufatura aditiva de concreto (*additive manufacturing os concrete – AMoC*). A impressão de argamassa funciona com o mesmo princípio da impressão tridimensional por deposição de material fundido (*Fused Deposition Modelling – FDM*).

A argamassa de cimento é um dos materiais mais utilizados na construção civil brasileira. O Brasil tem ampla experiência com o uso do concreto, o que permitiu uma grande variedade de inovações no campo da construção civil e na plasticidade apresentada pelos arquitetos brasileiros nas edificações construídas no século XX. A

argamassa cimentícia é um material de grande versatilidade que pode receber aditivos para melhorar sua *performance* e ser utilizado em diferentes propostas e métodos construtivos.

Existem poucos exemplos de impressoras utilizadas para a construção de edificações inteiras em tamanho natural, escala 1:1. Alguns exemplos como o projeto dos pesquisadores da ETH de Zurique (representado pela Figura 20), o projeto DFAB House da pesquisa de Gramazio & Kohler e o pavilhão Flotsam e Jetsam da empresa Branch Technologies.

Figura 20 – Edificação com elementos impressos – ETH de Zurique

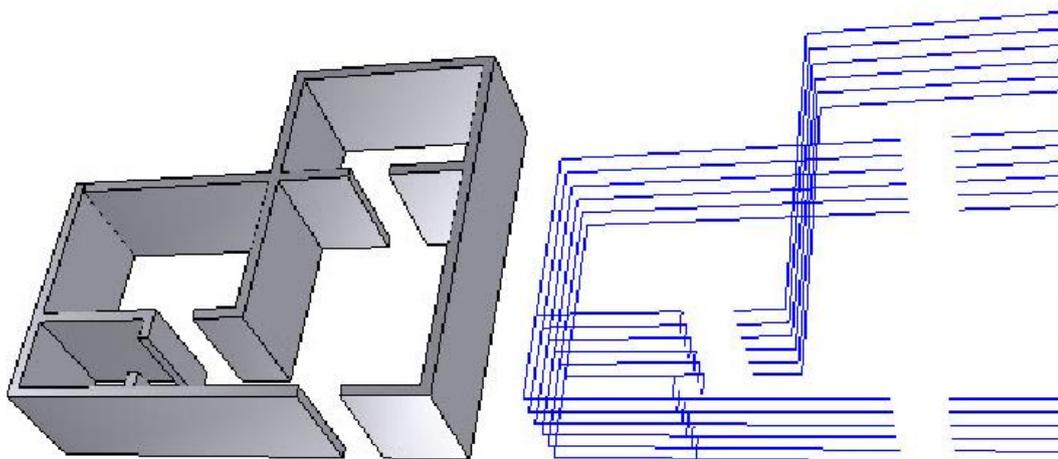


Fonte: ArchDaily, 2018.

O primeiro passo para a impressão 3D de edificações em escala 1:1 é desenvolver um modelo tridimensional utilizando um *software* de modelagem que permita salvar o arquivo em formato compatível com os *softwares* de gerenciamento de impressão. O formato de arquivo mais comum é o *stereolithography* (.STL).

O segundo passo é abrir o arquivo .STL em um *software* de fatiamento para salvar o código gerado para impressão. O código de gerenciamento de impressão ou código G (*G.Code*) consiste em uma série de coordenadas cartesianas que a programação da impressora reconhece e define como os eixos devem se mover, *vide* Figura 21:

Figura 21 – Projeto fatiado para impressão 3D



Fonte: Al-Azhari e Allouzi, 2020.

O código G gerado a partir do modelo virtual do projeto define os caminhos que devem ser percorridos pelos eixos da máquina, calcula a quantidade de material necessário para executar a impressão da edificação e o tempo necessário para o trabalho.

Na impressão de plástico, é importante que seja feita em apenas uma etapa para garantir maior qualidade do objeto. Contudo, na impressão de cimento, dependendo do traço utilizado, pode ser necessário pausar a impressão para que uma determinada quantidade de camadas passe pelo processo de cura do concreto, garantindo a integridade estrutural da edificação.

A impressão pode ser definida como *in site*, feito no local da edificação ou *out site*, feito em fábricas e transportadas para o local de montagem da edificação. A impressão de materiais pastosos consiste no bombeamento de argamassa de concreto ou terra armazenada em um reservatório por meio de um bico de extrusão acoplado a eixos de movimentação XYZ.

A liberdade de movimentação dos eixos de uma impressora 3D e as propriedades naturais e agregadas das argamassas definem as linhas guias para a criação de projetos (MARIJNISSEN; VAN DER ZEE, 2017).

2.12 Traços utilizados

O foco do estudo apresentado na tese não é a definição de traços para impressão 3D. Contudo, é interessante apresentar um exemplo de traço desenvolvido

por SG weber Beamix (BOS; WOLFS; AHMED; SALET, 2016). O estudo de Beamix apresenta algumas características interessantes para argamassas destinadas a impressão 3D.

Composição do cimento para impressão tridimensional (SG weber Beamix):

Argamassa:

- Cimento Portland (CEM I 52,5 R);
- Agregado silicoso com granulometria de no máximo 1 mm;
- Enchimento de calcário e aditivos específicos – para facilitar o bombeamento;
- Modificadores de reologia para obtenção de comportamento tixotrópico da argamassa fresca; e
- Pequena quantidade de fibras de polipropileno para redução de formação de rachaduras nos primeiros estágios de cura

A argamassa não tem deformação acentuada, o que permite uma definição clara entre a abertura do extrusor e do caminho de impressão.

Existe forte interdependência entre o projeto (*design*), material de impressão, o processo definido de impressão tridimensional e o produto projetado.

É importante ressaltar que, em processos convencionais de impressão 3D, as camadas são iguais e com tempo de secagem rápido e homogêneo. Contudo, no caso de impressão com argamassa de cimento e terra é relevante considerar que as camadas iniciais e subsequentes podem apresentar tempos de cura diferentes.

2.13 Custos da impressão em m²

Algumas iniciativas estão desenvolvendo edifícios com impressão 3D de argamassa cimentícia. Os projetos têm por objetivo entender o custo da construção civil com esse tipo de tecnologia. O custo de uma edificação construída com impressão 3D não pode ser definido da mesma forma de uma edificação convencional. Algumas pesquisas apontam que a impressão 3D reduz os custos

devido ao fato de diminuir a mão de obra e não precisar de formas para a moldagem do concreto (CHEN; CHUNG; LI; YANG, 2018).

Um exemplo é o projeto desenvolvido para o “Office of the Future” (cf. Figura 22) em Dubai. O projeto é um conjunto de edifícios comerciais fabricados em galpões com impressão 3D e transportados para o terreno. Os estudos vinculados ao “Office of the Future” mostraram que os custos com mão de obra foram reduzidos de 50% a 80% e o desperdício foi reduzido de 30% a 60%, comparados com os métodos tradicionais.

Figura 22 – Escritório do Futuro



Fonte: Archdaily, 2016.

Outro aspecto que apresenta vantagem para a redução dos custos é o tempo de fabricação. Uma impressora 3D de argamassa cimentícia pode imprimir edificações em pouco tempo. A empresa Apis-cor apresentou, no ano de 2016, uma edificação de 100 m² produzida em 24 horas (o tempo de impressão não engloba as instalações e acabamentos), conforme Figura 23:

Figura 23 – Apis Cor. Conjunto de casas Impressas



Fonte: ArchDaily, 2019.

Em 2015, a construtora WinSun alcançou a façanha de montar dez casas em menos de 24 horas (cf. Figura 24), valendo-se de uma impressora de 32 metros de largura por 6,50 metros de altura e 150,00 metros de comprimento.

Aleksandrovich et al. (2019) aponta uma redução de 25% do tempo de construção para edifícios térreos e 23% nos custos.

A fabricação de edificações em tamanho natural com impressoras 3D de materiais pastosos tem se mostrado um processo promissor para redução de tempo e custos. Pesquisas futuras devem estudar meios para integrar a tecnologia de impressão 3D em tamanho natural a sistemas automatizados, como braços robóticos e inteligência artificial, para melhorar o reforço estrutural e a execução das passagens para instalações elétricas e hidráulicas de uma edificação.

Figura 24 – Casas desenvolvidas pela empresa chinesa Winsun, em Suzhou/China, 2015



Fonte: Marijnissen e Van der Zee, 2017.

Os projetistas podem elaborar propostas específicas para cada região e para resolver problemas únicos dos moradores ou dos condicionantes bioclimáticos sem aumentar o custo das edificações, pois o sistema construtivo permite que a obra seja personalizada.

A primeira casa construída com tecnologia de impressão 3D em tamanho natural no Brasil está localizada no município de Macaíba, Rio Grande do Norte. Estudantes de engenharia civil da Universidade Potiguar (UnP) construíram uma impressora em tamanho natural e imprimiram uma casa em cimento. Segundo Santos (2020), a impressora e a casa foram construídas entre os anos de 2017 e 2020. A impressão da casa de 66m² durou 7 meses devido aos ajustes necessários para definição do traço da argamassa e os ajustes necessários para o funcionamento esperado para a máquina, exemplificado pela Figura 25 abaixo:

Figura 25 – Primeira casa impressa no Brasil. Universidade Potiguar/RN



Fonte: 3DHomeConstruction, 2020.

Os estudantes da Universidade Potiguar contaram com a ajuda do grupo de pesquisa da Pós-Graduação em Engenharia da Universidade Federal do Rio Grande do Norte (UFRN) para desenvolver o bombeamento da argamassa. A impressora montada na Universidade Potiguar é composta por um pórtico de estrutura metálica, treliças simples e espaciais, com medidas de 3 metros de altura, 7,60 metros de largura e 12 metros de comprimento.

2.14 Uso da tecnologia de fabricação no ensino de projeto e sua evolução

A tecnologia está provocando mudanças e rupturas importantes em diversos segmentos produtivos, gerando novas maneiras de criar, projetar, produzir objetos, equipamentos e edificações.

Centros de produção e fabricação digital (cf. Figura 26) surgiram no mundo todo idealizados pelo projeto FABLAB do Instituto de Tecnologia de Massachusetts (MIT).

Figura 26 – *Lab. Maker/maquetaria* – Instituição de Ensino Superior IESB. Brasília/DF, 2020



Fonte: o autor.

Os laboratórios de fabricação digital proporcionam um espaço adequado para incentivar a cultura do aprender por meio da prática. Os estudantes e profissionais podem elaborar suas propostas ou tentar responder a questionamentos levantados com experimentos e produção de modelos para estudo.

O espaço colabora para o desenvolvimento de ideias e o aprendizado de áreas do conhecimento vinculadas ao uso de equipamentos de marcenaria, robótica, eletrônica, fabricação digital, programação de ferramentas digitais e físicas, entre outras tecnologias que beneficiam formas de pensar e executar projetos.

Esses laboratórios são constituídos por grupos de pesquisadores que atuam em diferentes áreas ligadas à ciência ou tecnologia. Para isso, essas pessoas utilizam sua experiência, os conhecimentos e os projetos de construção/fabricação de equipamentos dos próprios membros do grupo ou aqueles tornados públicos por meio da internet (DEMETRIO; SAMAGAIA, 2015).

Nesse contexto, foi fundada a rede FABLAB (cf. Figura 27), que se caracteriza por redes abertas e colaborativas de laboratórios de fabricação digital. Ela está presente em vários países e foi iniciada pelo professor Neil Gershenfeld, diretor do Center for Bits and Atoms (CBA), do MIT (ANGELO et al., 2012).

Figura 27 – FABLAB. MIT *Media Lab*.



Fonte: *Media Lab*/MIT, 2021.

Os laboratórios de fabricação digital estão se transformando em ambientes cada vez mais necessários nas escolas de Arquitetura e Urbanismo, pois auxiliam o aprendizado baseado na experimentação. Os laboratórios criam a integração de dispositivos de *hardware* e *software*, que ajudam na criação de protótipos e artefatos arquitetônicos. Os espaços de fabricação digital mesclam tecnologias, tais como impressoras 3D e máquinas de corte computadorizados (CNC), com o objetivo de permitir que os estudantes entendam o processo construtivo e desenvolvam pesquisas de novos materiais e métodos construtivos.

Os estudantes podem desenvolver projetos em diversas frentes de estudo. Os trabalhos desenvolvidos nos laboratórios de fabricação digital visam melhorar alguma atividade cotidiana, de forma a integrar o raciocínio de projeto com a possibilidade de resolver os problemas com protótipos.

Um exemplo desenvolvido por pesquisadores e estudantes é o projeto *e-Nable*. Estudantes de cursos que envolvem habilidades de projeto utilizam *softwares* de modelagem digital para produzir próteses impressas para crianças. A proposta permite que estudantes resolvam problemas reais envolvendo a definição do programa de necessidades com as crianças e famílias, a produção e montagem das mãos e análise de uso.

Estudantes de Arquitetura e Urbanismo participam modelando tridimensionalmente o membro a ser fabricado digitalmente, ampliando o conhecimento sobre o sistema de prototipagem. A proposta possui o aspecto social

por ajudar sem custo pessoas que não poderiam comprar uma prótese convencional, como pode ser visto na Figura 28:

Figura 28 – Produção de prótese para crianças – estudante de arquitetura e voluntária do projeto *e-Nable*, IESB



Fonte: o autor.

As máquinas de fabricação digital auxiliam na produção de maquetes e modelos em escalas reduzidas, utilizando o mesmo método construtivo aplicado em edificações impressas em escala natural.

Esta tese apresenta a proposta de uma impressora projetada para extrudar argamassa e imprimir objetos de estudo e maquetes. A impressora fabrica modelos e artefatos arquitetônicos com a mesma técnica e o mesmo tipo de material da impressão em tamanho natural, aproximando o projetista e o projeto do resultado final esperado.

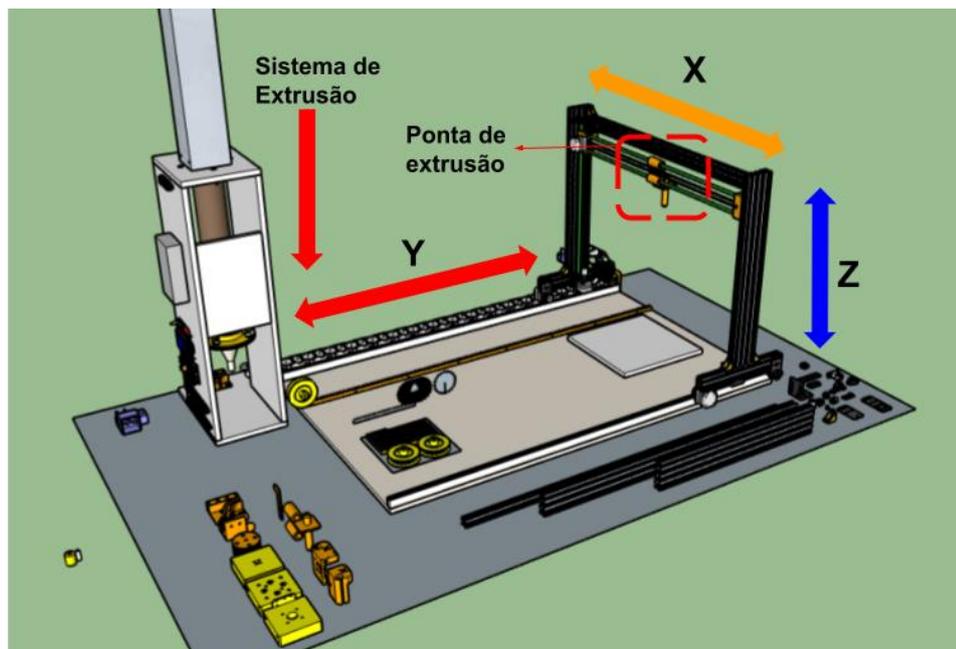
A impressora deve ser utilizada para apresentações em laboratório para permitir que os estudantes da FAU/UnB tenham um contato precoce com a tecnologia de manufatura aditiva.

3 Impressora 3D de extrusão de argamassa – PPG-FAU/UnB

A introdução de tecnologias de código aberto e plataformas de construção personalizadas é uma necessidade latente para a popularização e evolução das impressões de edificações. Contudo, é necessário aplicar conhecimentos nas áreas de utilização de *softwares* e *hardwares* (KONTOVOURKIS; TRYFONOS, 2018).

A tese apresenta a participação no desenvolvimento e construção de uma impressora 3D de argamassa, conforme apresentado na Figura 29. A máquina foi projetada com a colaboração de pesquisadores do Laboratório de Fabricação Digital e Customização em massa (LFDC) e EdiQuali, ambos da Faculdade de Arquitetura e Urbanismo da Universidade de Brasília (FAU/UnB).

Figura 29 – Projeto para impressora em escala para argamassa de terra



Fonte: EdiQuali, FAU/UnB, 2020.

A impressora opera utilizando o sistema cartesiano de movimentação em eixos X e Y e a dimensão tridimensional é proporcionada pelo movimento vertical da ponta de extrusão, eixo Z.

A impressora foi projetada para testes de impressão com materiais pastosos, tais como argila, argamassa de cimento e de terra. Sua estrutura de alumínio possui um pórtico que se movimenta no eixo Y (*vide* Figura 30).

Figura 30 – Construção de impressora em escala para argamassa de terra



Fonte: EdiQuali, FAU/UnB, 2020.

A construção e utilização da máquina permite simular uma impressora de tamanho natural para entender como o material deve ser extrudado, suas características físicas e tempo de impressão.

A montagem da impressora permitiu observar como a programação influencia o funcionamento e quais serão os parâmetros mais relevantes para o funcionamento da máquina. A impressora pode ser utilizada para execução de testes de movimento e extrusão, contribuindo para o desenvolvimento de algoritmos, ferramentas digitais e no ensino de projeto.

A construção da máquina foi de grande relevância para o estudo. O aumento significativo de proporções apresentou como os eixos se comportam em velocidades diferentes e a necessidade de calibrar o movimento dos motores de acordo com as medidas definidas para a área de impressão. O modelo com um pórtico que se movimenta apresentou como deve ser a calibragem da velocidade de impressão para não criar trepidações ou problemas para interromper o movimento devido à quantidade de movimentos do pórtico ser maior por ter maior massa.

3.1 Programação da impressora – PPG-FAU/UnB

A impressora 3D desenvolvida na FAU/UnB utiliza como base o sistema RepRap com a controladora de uso geral Arduino mega 2560 e o conjunto de programação (*firmware*) *open-source Marlin*.

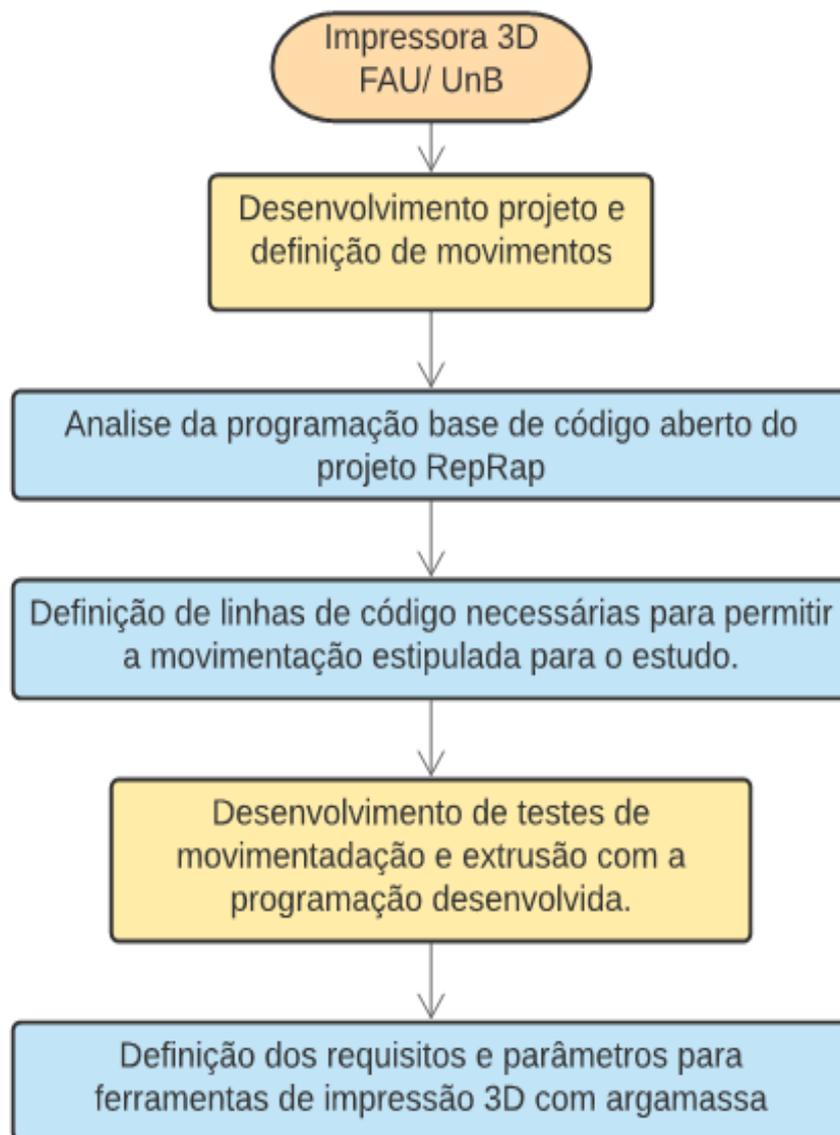
A máquina utiliza uma programação particular para o *hardware* (*embedded control systems*) para definir o seu funcionamento, movimento de eixos, início e término do trabalho. A impressora teve sua programação adaptada para as necessidades estipuladas para simular uma impressora de tamanho natural de materiais pastosos.

A programação utilizada na impressora é publicada na internet de graça com permissão do tipo Licença Pública Geral (GNU, na sigla em inglês). O código da impressora é escrito no Ambiente de Desenvolvimento Integrado (IDE) da placa controladora Arduino. A IDE Arduino é um programa de código aberto e gratuito que pode ser utilizado em diferentes sistemas operacionais.

O texto apresentado no código da impressora com duas barras inclinadas (//) serve para que o IDE do Arduino não interprete as linhas como programação. São textos utilizados para que o autor do programa ou outros usuários possam entender o que foi feito ou mesmo modificado.

3.1.1 Fluxograma para desenvolvimento da programação da impressora

Aqui, são apresentados a programação para a impressora da FAU/UnB e os requisitos definidos para seu funcionamento durante o processo de impressão. O projeto foi desenvolvido em conjunto com o professor Dr. Márcio Albuquerque Buson, que fez o modelo digital da impressora no *software Sketchup*. As peças da impressora foram confeccionadas com uma impressora 3D do tipo FDM.



3.2 Parâmetros para impressão 3D de argamassa

Nos projetos de arquitetura, todo desenho proposto tem uma relação direta com o material e com o sistema construtivo adotado. Fica evidente a relação entre o material escolhido para a obra, os elementos estruturais especificados, o desenho da edificação e a técnica construtiva adotada (HEISE; HOFFMANN; MINTO, 2011).

É relevante considerar alguns parâmetros característicos das argamassas e materiais pastosos. As argamassas apresentam o índice de plasticidade, definido como a diferença entre o limite de liquidez e o limite de plasticidade ($IP = LL - LP$). O

grau de umidade presente no solo vai definir se é um solo líquido, plástico ou sólido (ABNT, 1995).

A retração é um parâmetro ligado à quantidade de argila ou materiais argilosos presentes no solo, que são responsáveis pelos movimentos de retração e expansão. O limite de retração (LR) é determinado pelo grau de umidade presente na argamassa. Além disso, ela é um parâmetro a ser incorporado em um algoritmo para auxiliar no ajuste do modelo virtual. A indicação do abatimento desejado e o material de estabilização podem modificar o arquivo de impressão e o comportamento do material impresso.

3.2.1 Blocos de algoritmos para definição de requisitos de impressão 3D de argamassa

É relevante definir blocos de algoritmos para agilizar o desenvolvimento de ferramentas para impressão tridimensional com argamassas. Os blocos devem ser direcionados para propiciar o desenvolvimento das ferramentas por arquitetos e profissionais de programação:

- Blocos para modificação de medidas de milímetros para centímetros e metros;
- Blocos para definição do tamanho do bico de impressão;
- Blocos para pausas e retomadas de impressão com materiais pastosos;
- Blocos para controle de velocidade da impressão com materiais pastosos;
- Bloco para definição de alturas de camadas relacionado ao parâmetro de retração;
- Bloco relacionado ao parâmetro de plasticidade das argamassas; e
- Definição de requisitos de uso para impressão 3D em tamanho natural.

Os blocos são apresentados em Português, “português estruturado”, sistema de programa de fácil entendimento e que pode ser utilizado para desenvolver ferramentas digitais em diferentes linguagens de programação.

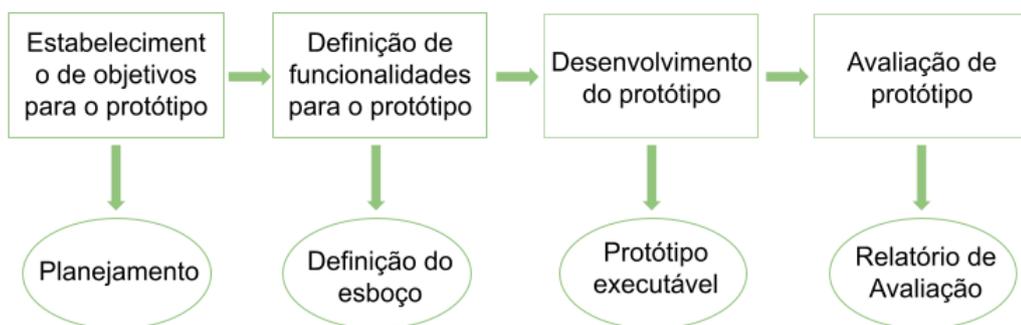
Os algoritmos são apresentados em formato de fluxograma nos apêndices do trabalho. Os fluxogramas permitem o entendimento da sequência de funcionamento

dos códigos. O sistema em fluxo facilita o desenvolvimento de propostas e projetos de programas.

4 Programação e parâmetros – *softwares open-source*/livres

É relevante definir quais etapas de desenvolvimento para o protótipo do algoritmo serão necessárias. É importante desenvolver de forma rápida as partes do sistema e verificar sua relevância para a pesquisa, ou seja, criar blocos de códigos com os parâmetros definidos para a impressão em argamassa e movimentação dos eixos da impressora (SOMMERVILLE, 1996). *Vide* Figura 31:

Figura 31 – Processo de desenvolvimento de protótipo



Fonte: *Software Engineering*, Sommerville, 1996.

O programa a ser desenvolvido/modificado é um programa personalizado para a utilização de impressoras tridimensionais de projeto aberto (*open-source*). É relevante que o programa seja de fácil entendimento e de funcionamento simplificado. A ferramenta é dirigida para projetistas e construtores.

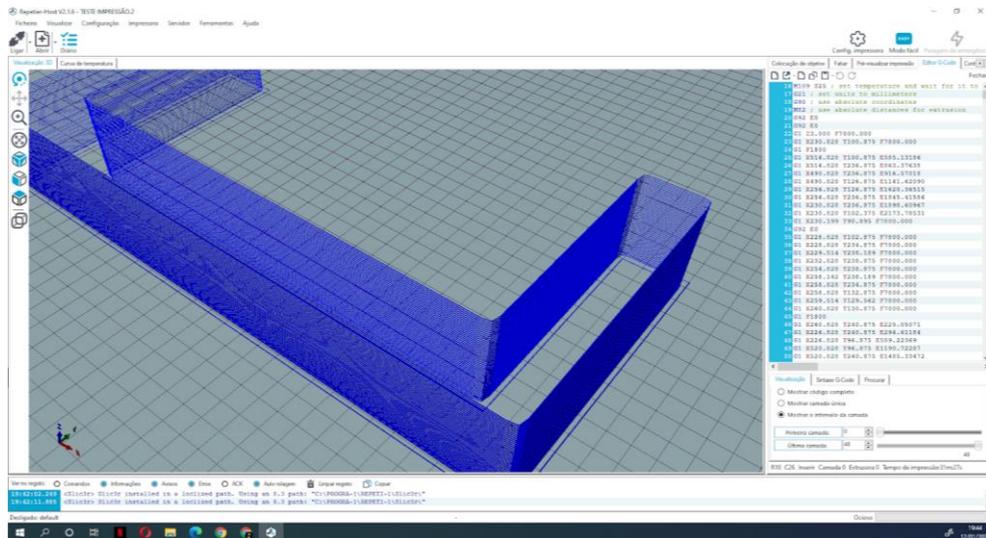
O uso de *software* livre permite aos usuários executar, copiar, distribuir, estudar, alterar e modificar programas. É necessário que o programa seja liberado sob uma licença para *software* livre como a Licença Pública Geral (GNU).

As máquinas de controle numérico computadorizado, como máquinas de corte e impressoras 3D, utilizam programas que transformam projetos 2D ou 3D em coordenadas cartesianas. Os programas para impressão 3D são chamados de fatiadores, por transformarem os modelos 3D em camadas e converterem os arquivos para o formato *Geometric Code*, *G-code* ou código G, que comandam a impressora indicando o que deve ser feito, como se mover e quais caminhos os eixos devem percorrer (DEJAN, 2021).

Na Figura 32, é demonstrado um exemplo de programa livre, *Repetier-host*, e um modelo fatiado com seu código G ao lado. O código G é uma sequência de

coordenadas cartesianas que indicam os movimentos bidimensionais e como a impressora deve movimentar os eixos X, Y e Z para criar o objeto 3D.

Figura 32 – Simulação de impressão 3D - Programação G-code



Fonte: Autor.

Exemplos de alguns comandos importantes referentes ao código G:

- **G 00** // mudança de posição rápida para um ponto determinado. Movimento sem ação;
- **G 1** // move em linha reta em um ponto específico. É relacionado à movimentação do trabalho principal da máquina. Em impressoras 3D, é o movimento ligado à extrusão de material.

Exemplo: **G 1 X230.828 Y100.875 F7800.000**

G 1 // movimento Linear;

X 230.828 e Y100.875 – Definem a posição final do movimento;

F 7800.000 – Velocidade do movimento. A velocidade do exemplo está em mm/s.

G2 // movimento circular em sentido horário;

G3 // movimento circular em sentido anti-horário;

G17 // define o plano de trabalho da máquina – XY plane;

G21 // definição de sistema métrico; e

G28 // retorna para a origem dos espaços (*Home*) – É possível inserir pontos intermediários de movimentação para evitar colisões.

A possibilidade de colisões da extrusora com o artefato impresso foi cogitada no projeto da impressora 3D FAU/UnB, e, para evitar acidentes, o eixo Z termina seu movimento subindo para a parte superior do pórtico.

G 90 // modo absoluto – O posicionamento da ferramenta é a partir do ponto absoluto ou zero. Ou seja, a máquina de corte ou impressão 3D movimenta os eixos para o ponto exato indicado na programação do código G; e

G91 // modo relativo – A máquina vai retomar o movimento utilizando as coordenadas relativas ao posicionamento dos eixos, somando as coordenadas.

A programação código G é vasta e apresenta diversas variáveis de comando. Os mais significativos estão apresentados acima para ilustrar o seu funcionamento. O desenvolvimento da ferramenta de impressão de argamassa precisa utilizar diversos outros comandos que serão apresentados na tese.

A programação código G é complementada pelo código M para máquina de corte e impressoras 3D. No caso de máquinas de corte com motor *Spindle*, motor de alta frequência para cortes com lâminas fresas, é necessário completar a programação com o código S.

5 Programas *open-source* para análise

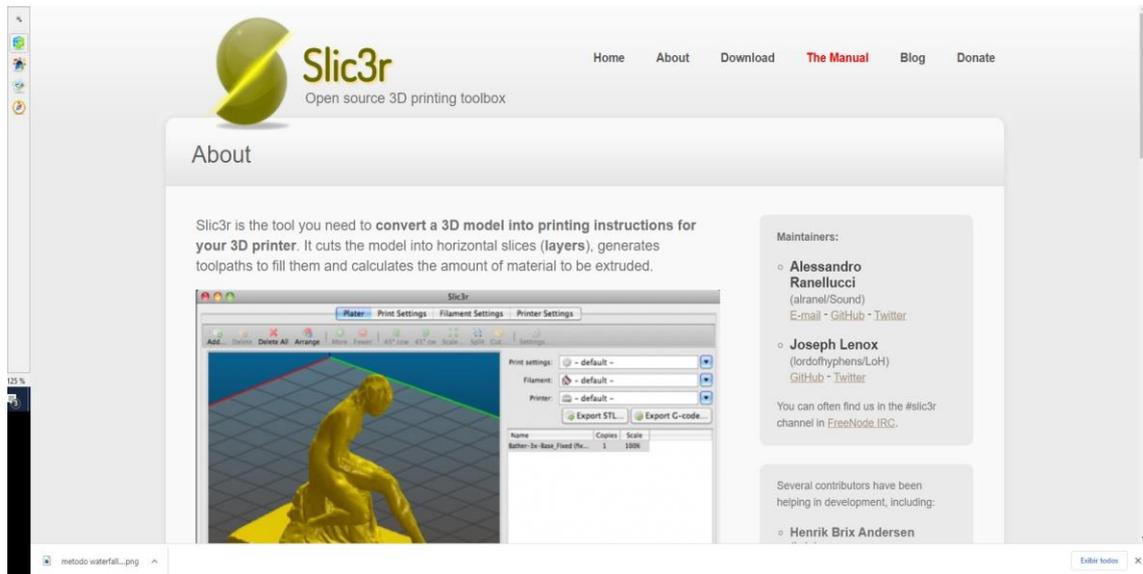
O desenvolvimento da ferramenta de impressão 3D de argamassa tem por objetivo utilizar e modificar programas livres preexistentes. O uso dessas ferramentas livres permite um desenvolvimento ágil por utilizar um código disponibilizado na internet pela comunidade de desenvolvedores.

O principal sistema escolhido para o projeto é o *reuse-oriented software engineering*, engenharia de *software* orientada para reutilização, que é baseado na existência de um número significativo de componentes reutilizáveis. O sistema é focado em integrar diversos componentes em vez de começar um projeto do zero. O *reuse-oriented software engineering* é um sistema criado há mais de 40 anos, contudo, apenas por volta dos anos 2000 se tornou o padrão para o desenvolvimento de novas ferramentas digitais. O movimento *open-source* apresentou uma grande base de códigos para reuso a baixo custo ou nenhum custo, que pode ser apresentada na forma de bibliotecas de programação ou em programas completos (SOMMERVILLE, 1996).

Esse tipo de desenvolvimento de *software* é um dos mais interessantes para o trabalho proposto, pois apresenta como blocos de códigos ou códigos completos podem ser reutilizados para o desenvolvimento de novas ferramentas. O sistema de reuso de algoritmos tem a vantagem de reduzir a quantidade de código desenvolvido, diminuindo custos e riscos de desenvolvimento.

Os algoritmos relacionados a impressão de argamassa que são apresentados na tese são liberados na internet para que a comunidade acadêmica e de desenvolvedores possam utilizar ou modificar. O desenvolvimento dos códigos utiliza como base os *softwares* criados para impressoras de deposição de plástico fundido tipo *desktop*: *Slic3r* e *Printrun/Pronterface*. Veja-se a Figura 33:

Figura 33 – Programa de código aberto *Slic3r*



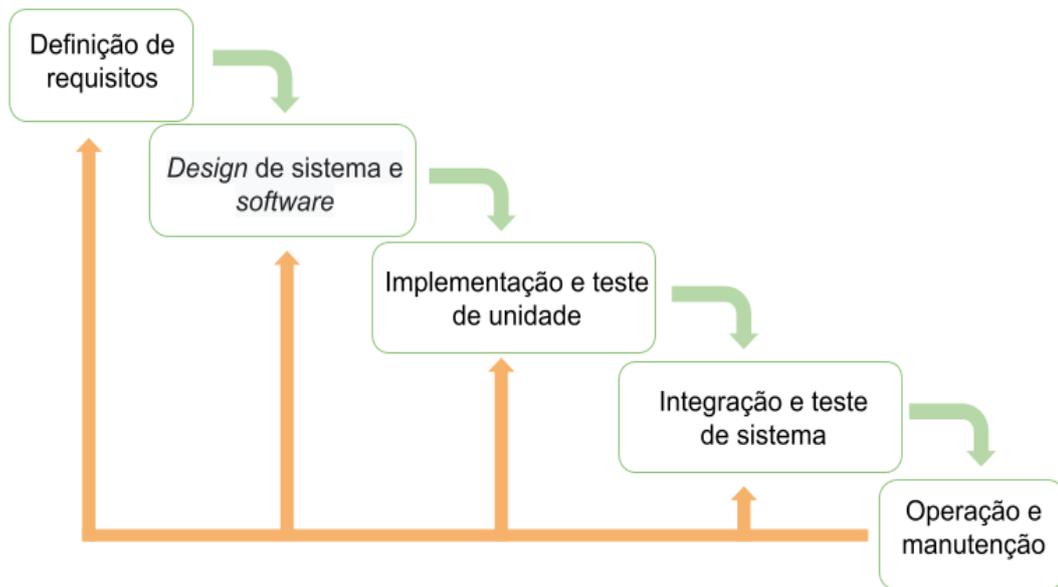
Fonte: *Slic3r*, 2021.

Os dois *softwares* selecionados para o estudo são escritos em linguagens de programação diferentes. O programa *Slic3r* utiliza a linguagem de programação C++, e o *Printrun/Pronterface* utiliza a linguagem *python*, linguagem largamente utilizada para diferentes ferramentas digitais.

O desenvolvimento de programas é composto por diversas etapas. Um programa pode ser desenvolvido do zero. Contudo, o desenvolvimento de *softwares* para negócios e aplicações variadas é normalmente efetuado por meio da modificação ou introdução de blocos de códigos em um programa preexistente.

A organização de *softwares* denominada *waterfall* é um dos sistemas abordados para o desenvolvimento da ferramenta proposta. É um método dirigido por planejamento e que necessita da elaboração de plano e cronograma de todas as atividades antes de começar a elaboração do algoritmo. Esse modelo de desenvolvimento pode ser visto na Figura 34:

Figura 34 – Modelo de desenvolvimento de *softwares Waterfall*



Fonte: Software Engineering, Sommerville, 1996.

Para o desenvolvimento dos algoritmos, é importante analisar os códigos dos programas *open-source* escolhidos.

Os códigos foram definidos com o objetivo de contribuir para o desenvolvimento de ferramentas digitais para impressão 3D, para facilitar o uso de impressoras tridimensionais de tamanho natural e para contribuir no controle adequado do processo de extrusão de material pastoso.

5.1 Algoritmos e requisitos para definição de ferramenta de impressão 3D

Algoritmos são conjuntos de instruções que orientam o computador, outros tipos de máquinas ou até mesmo pessoas a como realizar tarefas. Cada trecho de um código de um programa de computador pode ser chamado de algoritmo (Bhargava, 2017). Para Ventavoli (2014, p. 8),

Algoritmos: especificação de uma sequência de passos, ordenação que visa atingir um objetivo definido; Segue um determinado padrão de comportamento, com objetivo de alcançar a solução de um problema. É a descrição de um problema. São muito comuns no nosso cotidiano.

Um programa de computador tem a função de resolver um problema ou otimizar uma ação desempenhada por máquinas ou pelo ser humano. Os algoritmos são

definidos por um conjunto de ações ordenadas com a finalidade de chegar à solução de um problema (MILETTO; NICOLAO; OKUYAMA, 2014).

Os algoritmos podem ser divididos em algoritmos naturais e algoritmos computacionais (Ibidem, 2014). Algoritmo natural é um conjunto de passos necessários para cumprir com uma atividade específica. Um bom exemplo é uma receita de bolo ou a troca de um pneu, pois são atividades que necessitam de uma sequência lógica. O algoritmo natural pode ser escrito de diversas formas diferentes e mesmo assim chegar na mesma solução.

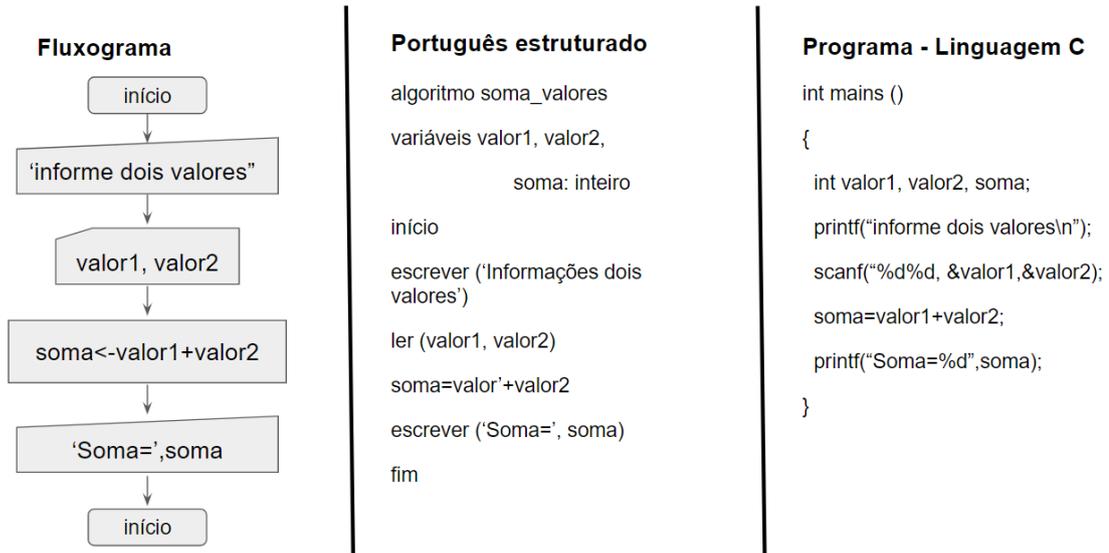
O algoritmo computacional requer um detalhamento mais próximo da linguagem de computador, ou seja, obedece a normas de sintaxe e semântica para que as ações indiquem comandos.

É possível representar algoritmos de diferentes formas, que devem ser definidas de acordo com a necessidade, etapa de desenvolvimento, com o problema ou para quem fará uso da representação. As formas mais comuns são o fluxograma e o “português estruturado”, também conhecido como “Portugol”.

Um algoritmo em português estruturado deve ser nomeado com a palavra **algoritmo**, seguida do nome escolhido pelo programador e pela seção de declaração de variáveis e constantes.

O português estruturado (*vide* Figura 35) é um pseudocódigo escrito em língua portuguesa em que as instruções definidas para o desenvolvimento de uma ferramenta digital podem ser entendidas por qualquer programador de forma simples e ágil e com a vantagem de não ser necessário um conhecimento prévio de alguma linguagem específica de programação como o C++ ou o *python*.

Figura 35 – Fluxograma, português estruturado e programa na linguagem C



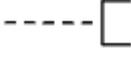
Fonte: Miletto, Nicolao e Okuyama, 2014.

O fluxograma é uma representação gráfica que auxilia no desenvolvimento de um programa e também pode ajudar no entendimento de como os blocos de programação funcionam (MILETTO; NICOLAO; OKUYAMA, 2014). O sistema de representação por meio de fluxogramas é uma forma mais amigável de apresentar o sistema de um *software* ou a explicação de funcionamento de um programa para um público não especializado, como arquitetos e projetistas. O fluxograma de desenvolvimento de ferramentas digitais segue regras para sua elaboração, como poderá ser visto na Figura 36.

O estudo apresenta fluxogramas e “português estruturado” como forma de desenvolver e apresentar as modificações definidas como necessárias para as ferramentas de código aberto.

Um código deve ser composto por diversos elementos. Os “Identificadores” são nomes dados às variáveis e devem ser escritos inicialmente com letras de “a” a “z” ou de “A” a “Z”. Na nomeação dos identificadores, não existe distinção para maiúsculas e minúsculas. Contudo, não é obrigatório que um algoritmo tenha variáveis ou constantes. Os blocos de comandos dos algoritmos devem começar com as palavras início e fim.

Figura 36 – Principais símbolos do Fluxograma.

Símbolo	Significado
	Início e fim do fluxograma
	Comando de atribuição
	Entrada de dados
 	saída de dados
	Decisão
	Seta de fluxo de controle
	Conector de linhas de fluxo
	Comentário
	Conector de fluxo em outra página

Fonte: Miletto, Nicolao e Okuyama, 2014.

Os primeiros símbolos posteriores às letras dos títulos dos “identificadores” podem ser números ou ainda o *underscore* (_). Não é permitido utilizar outros caracteres especiais como %, @, #, acentos e palavras reservadas⁴. Para os algoritmos apresentados na tese, os nomes são definidos começando com a palavra “algoritmo”, em letras minúsculas, e seguido do nome da função ou ação definida pelo código em letras maiúsculas. Exemplo:

algoritmo CONTROLE_VELOCIDADE.

Outro componente importante para os algoritmos e requisitos para definição de ferramenta são as “variáveis”, partes importantes dos programas, responsáveis por armazenar informações. A nomeação das variáveis segue as mesmas regras

⁴ “Palavras reservadas” são comandos que servem para executar instruções. Cada linguagem possui suas próprias palavras reservadas, que não podem ser utilizadas como “identificadores”, por serem reservadas para uso específico na linguagem (MILETTO; NICOLAO; OKUYAMA, 2014).

definidas para os identificadores. O tipo de dado determina o conjunto de valores que uma variável pode armazenar. Para os blocos de algoritmos, teremos diversas variáveis relacionadas aos parâmetros da impressão tridimensional em tamanho natural. Exemplo de variável:

Tipo: Real.

Descrição: VALOR NUMÉRICO COM PONTO DECIMAL.

Exemplo: 1.00, 5.60, 12.00.

O algoritmo apresenta uma seção de declaração de variáveis, prática necessária no desenvolvimento de qualquer programa. A declaração de variáveis começa com a palavra “variavel”, sem acento, pois é necessário utilizar a regra de nomeação dos identificadores, seguida das declarações de cada variável:

Formato de algoritmo para declaração de variáveis:

<identificador [,identificador...]>:<tipo>

Variáveis do mesmo tipo podem ser declaradas umas ao lado das outras separadas por vírgula. Exemplo:

MEDIDA, BICO_EXTRUSOR, ESPESSURA_CAMADA: real

É necessário atribuir valor às variáveis indicadas nos algoritmos. O valor é indicado tanto em Portugol quanto em fluxograma.

Exemplo Portugol:

```

1  algoritmo MODELO_IMPRESSORA (cf. Figura 37);
2  // Seção de declaração de variáveis;
3  variavel BASE_IMPRESSAO_X:real
5  variavel CUSTOM_MACHINE_NAME:caracter;
6  inicio // identificação do início do bloco correspondente ao algoritmo;
7  // corpo do algoritmo – lista de comandos – local designado para inserção de
todas as instruções que o programa deve realizar, entrada – processamento – saída
de dados;
8  escreva ‘Pos–UnB.08.06’;
```

9 **fim** // fim do algoritmo.

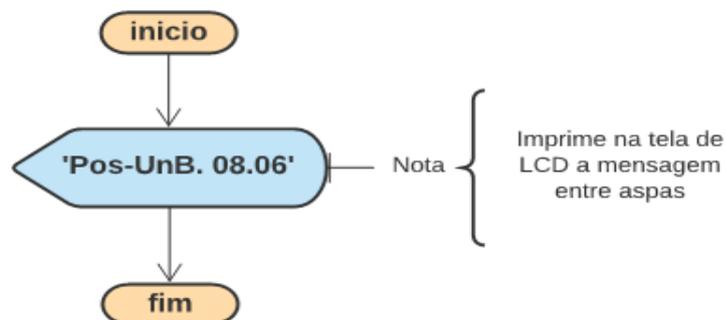
Figura 37 – Tela de comando da impressora projetada e construída na FAU/UnB



Fonte: Autor.

A Figura 38 apresenta exemplo de fluxograma para imprimir a mensagem Pos-UnB.08.06 na tela:

Figura 38 – Algoritmo para apresentar na tela LCD a mensagem 'Pos-UnB. 08.06'. Definido em 08/06/2020



Fonte: Autor.

A palavra “Pos” não tem acento pois o visor de *Liquid Crystal Display* (LCD) utilizado na impressora da FAU/UnB não reconhece esse tipo de caractere.

Os algoritmos devem seguir regras simples de construção:

- Devem apresentar um enunciado;
- Usar um verbo por frase;
- Apresentar frases simples e curtas. Deve ser escrito de forma objetiva;
- Não utilizar palavras com sentido duplo; e

- O algoritmo deve ser o mais simples possível.

Os códigos devem ser construídos em etapas, com o entendimento dos problemas abordados e da definição dos requisitos de funcionalidade. É necessário identificar e manter separadas as entradas, o processamento e a saída de dados. O algoritmo deve seguir uma sequência lógica de comandos.

As linguagens de programação possuem palavras reservadas que não podem ser usadas para outro propósito que não seja previsto nas regras de sintaxe. As palavras reservadas são comandos predefinidos nas linguagens de computador que são interpretadas pela máquina. A palavra reservada deve ser utilizada de forma correta para que a sequência lógica dos algoritmos seja reconhecida pela máquina.

6 Programa *PRONTERFACE/PRINTRUN* – requisitos para especificação de *software*

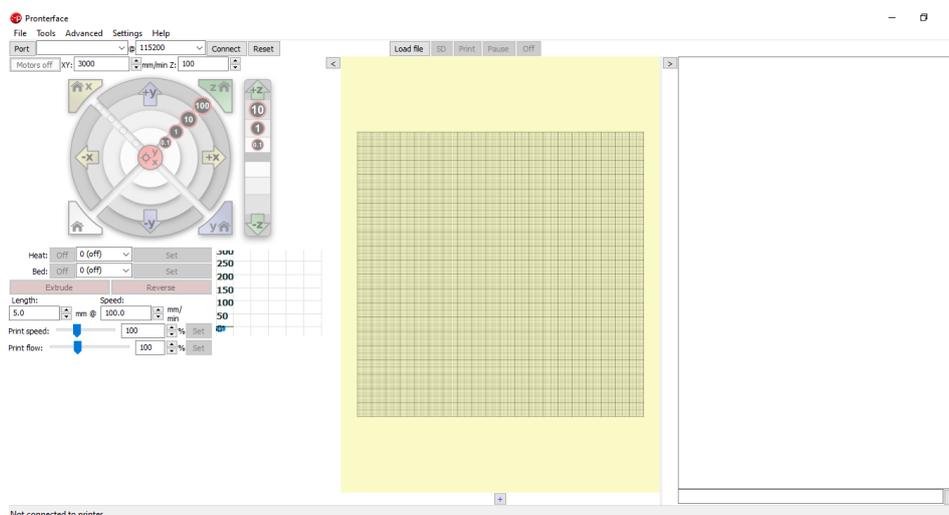
Um dos programas escolhido para servir de base para o desenvolvimento dos algoritmos e parâmetros para a impressão de argamassas foi o *software* livre *Printrun/Pronterface*, plataforma de interface para impressoras 3D e máquinas de corte numérico computadorizado (CNC).

A programação da ferramenta é feita com a linguagem *Python*. É uma linguagem de propósito geral de alto nível, ou seja, mais próxima da língua utilizada pelos seres humanos. *Python* é uma linguagem orientada a objeto, visa aproximar a programação do mundo real e os desenvolvedores podem criar ferramentas que definem um objeto ou uma ação. A linguagem foi lançada em 1991 e possui um modelo de desenvolvimento comunitário, aberto e gerenciado pela associação sem fins lucrativos de mesmo nome.

A linguagem *Python* foi um dos requisitos que orientou a escolha do programa *Printrun/Pronterface*. Programas escritos em *Python* devem ser dinâmicos, bem comentados e devem permitir a fácil leitura do código por outros usuários. A programação deve ter poucas linhas de código quando comparada a programas similares escritos em outras linguagens.

O programa permite que o usuário faça o processo de movimentação dos eixos da impressora por meio de um controlador virtual, visualização do arquivo 3D e visualização prévia das camadas e padrões de preenchimento, como poderá ser visto na Figura 39. A ferramenta possibilita que os arquivos de impressão em código G sejam salvos em cartões de memória *Secure Digital* (SD) ou coordena a impressão direto por meio de sua interface gráfica.

Figura 39 – Interface de uso do programa *Printrun*/*Pronterface*.



Fonte: Autor

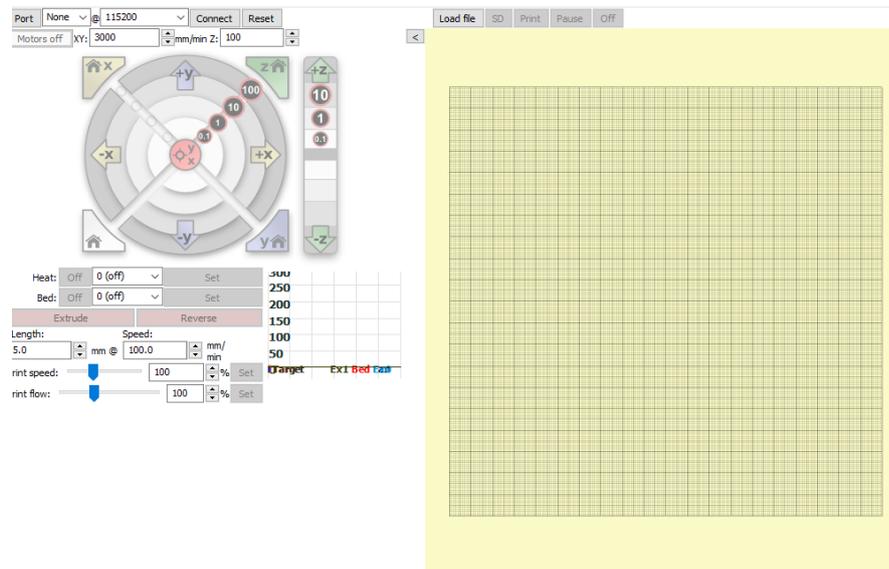
O *software* é licenciado sob o sistema *GNU Public License*, permitindo a personalização e modificação do código fonte. Todos os códigos relacionados ao programa devem conter uma licença disponibilizada no *site* que hospeda o programa:

*Copyright (C) 2011-2020 Kliment Yanev, Guillaume Seguin
Printrun is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. Printrun is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with Printrun. If not, see <<http://www.gnu.org/licenses/>>.*

Fonte: Kliment Yanev, Guillaume Seguin. Disponível em: <https://github.com/kliment/Printrun>.

A impressão feita diretamente do *software* é uma necessidade para a impressão em tamanho natural, ou seja, de edificações em escala 1:1. A ferramenta deve permitir que o operador da impressora tridimensional de argamassa possa parar a impressão e diminuir a velocidade de movimento dos eixos durante a produção da edificação. A necessidade de controlar diversos aspectos da impressão durante o trabalho é uma característica importante pois a construção/fabricação de uma edificação é composta por muitas etapas com diferentes tipos de componentes como tubulações para passagens de cabos ou tubulações hidráulicas. *Vide* a Figura 40:

Figura 40 – Controlador Virtual de eixos e base de impressão em milímetros



Fonte: o autor.

A movimentação dos eixos deve ser controlada e lenta para permitir que o operador acompanhe a extrusão da argamassa. O usuário deve garantir por meio dos comandos na tela que as primeiras camadas não tenham falhas para não acumular erros quando forem posicionadas umas das outras. O controlador deve evitar camadas deslocadas mesmo que poucos milímetros pois podem comprometer a estabilidade estrutural da edificação.

6.1 Blocos para modificação de medidas

Os *softwares* de impressão tridimensional disponíveis para o uso, livres ou pagos, são desenvolvidos para a impressão com máquinas tipo *desktop* e, portanto, suas medidas são todas em milímetros. Decorre daí a necessidade de adaptar as dimensões de entrada e saída no programa, para que seja possível a fabricação de edificações em escala natural.

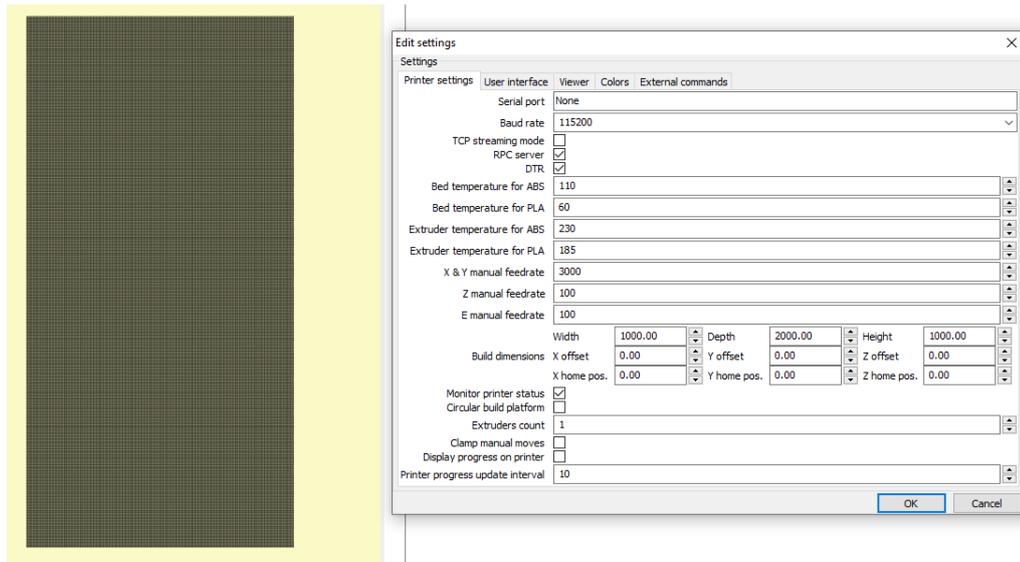
6.1.1 Bloco para medida e unidade da base – “*build dimensions*”

A ferramenta pode facilitar o uso por pessoas com formação em arquitetura e engenharia civil se for definida em metros. Os profissionais e estudantes relacionados

à construção civil estão acostumados a pensar em metros e centímetros devido às proporções dos objetos de estudo, das edificações e parcelas urbanas.

Na Figura 41, é possível observar o menu de configuração de medidas para base de impressão e da altura de uma impressora tridimensional. É necessário modificar os parâmetros “*width*” (largura), “*depth*” (profundidade) e “*height*” (altura).

Figura 41 – *Printrun/Pronterface*. Definição de tamanho de área de impressão



Fonte: o autor.

Os valores padrão do programa são 200x200x100+0+0+0+0+0+0, o que corresponde a 200x200 mm para área de impressão e 100 mm para altura de impressão. É necessário modificar o visualizador para metros, unidade de medida usual para o desenvolvimento de projetos de edificações.

O visualizador deve ser modificado para permitir uma maior interação com arquitetos e evitar que a base se “transforme” em uma mancha cinza, como observado na Figura 42, quando forem inseridos modelos em tamanho natural.

```
1 algoritmo BASE_METROS;

// Apresenta a modificação das características das bases de impressão. O algoritmo
// visa a indicação da base em metros e é subdividida em centímetros. O código
// apresenta linhas comentadas para que possa ser utilizado na impressora de estudo
// da FAU/UnB ou para uma impressora em tamanho natural.

2 variaveis unidade X, unidade Y: inteiros; // declaração de variáveis.

3 grade ← (1 cm; 10cm)); // definição de grade dividida em centímetros 4
// para impressora FAU/UnB de medidas reduzidas.

4 grade ← (10 cm ; 100cm)); // definição de grade dividida em
// centímetros/metros para utilização em impressoras de tamanho natural.

5 dimensões_base: inteiros;

6 inicio // identificação de início do bloco correspondente ao algoritmo.

7 unidade X ← 63;

8 unidade Y ← 89;

9 // medida da impressora de base de impressão para FAU/UnB.

10 // unidade X ← 600.

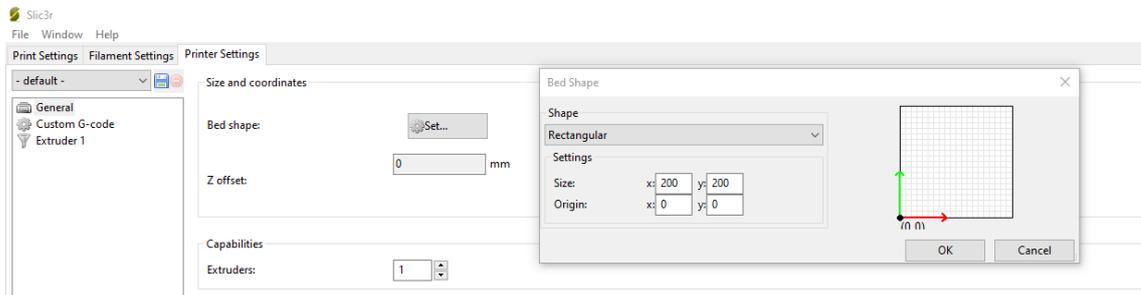
11 // unidade Y ← 900; // medidas da impressora para tamanho natural.

12 dimensões_base (unidade X, unidade Y, 0, 0, 0).

13 fim; // fim do algoritmo proposto.
```

O programa *Slic3r*, fatiador de código aberto inserido no *Printrun/Pronterface*, possui um menu de configuração da impressora que permite mudar o tamanho da base de impressão (cf. Figura 42). As medidas relacionadas à base dentro do *Slic3r* devem ser configuradas em metros para garantir a compatibilidade entre o fatiador e o programa de gerenciamento de impressão, *Printrun/Pronterface*.

Figura 42 – Slic3r. Menu de configuração do tamanho da base de impressão



Fonte: o autor.

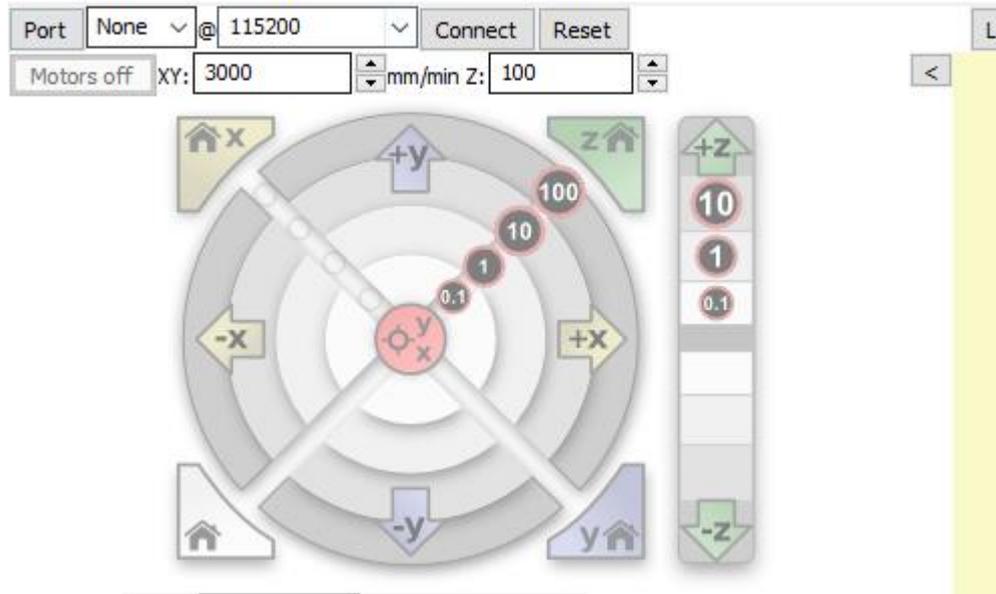
6.1.2 Bloco de algoritmo para medidas e parâmetros de controlador virtual

O controlador virtual original da ferramenta *Printrun/Pronterface* permite que o usuário movimente os eixos da impressora com medidas predefinidas em milímetros, como mostrado na Figura 43. O controlador virtual é uma ferramenta importante para uma impressora tridimensional de edificações em tamanho natural.

Em impressoras de mesa, a dimensão do aparelho permite que o usuário movimente os eixos com a mão sem danificar a máquina e o processo de impressão em pequenas bases faz com que seja simples e sem esforço. A movimentação dos eixos em uma impressora de grande porte só é possível quando controlado por meio da ferramenta. O usuário não deve movimentar manualmente as partes da impressora de edificações devido às suas grandes dimensões e peso. Os movimentos fora do momento de impressão são importantes para ajudar na configuração da impressão ou para organizar a rotina de trabalho.

Outro aspecto é o fato de que os objetos produzidos são edificações em escala 1:1. A edificação impressa cria obstáculos para a equipe responsável pelo funcionamento adequado do maquinário.

Figura 43 – Controlador virtual – eixos X, Y e Z



Fonte: o autor.

Os eixos X e Y devem ser definidos em metros e subdivididos em centímetros. O eixo Z também deve ter medidas em metros e em centímetros para permitir recolocar o eixo de extrusão em uma camada específica ou mesmo movimentá-lo em toda a extensão da altura da edificação ou altura do pórtico de impressão. O controlador possui representação gráfica denominadas “casinhas”, que indicam o ponto zero de cada eixo; no canto inferior esquerdo do controle virtual existe o comando para zerar todos os eixos.

O eixo Z, que define a tridimensionalidade do processo por meio da produção de sucessivas camadas sobrepostas, deve ter sua unidade principal modificada para centímetros e, assim, permitir o controle preciso quando for necessário reposicionar o extrusor.

```
1  algoritmo MOVIMENTO_Z

// O algoritmo foi desenvolvido para facilitar a movimentação do eixo Z. O código
// apresenta a possibilidade de inserção de velocidade e valores pré-definidos de
// espaços percorridos. Velocidade e distâncias são definidas em cm/min. O eixo deve
// ser em centímetros para permitir a colocação precisa do eixo Z na altura de camada
// desejada.

2  variaveis velocidade, valores_movimento: real;
3  multiplicador_distancia: inteiro;
4  constante home: logico;
5  inicio
6  escreva ('Insira a Velocidade Desejada');
7  leia (velocidade);
8  escreva ('velocidade=   cm/min', velocidade);
9  valores_movimento (0.5, 5, 50); // unidades de medida em cm.
10 multiplicador_distancia (0.5 * 10); // multiplica a unidade básica de medida 10
// para aumentar a distância percorrida.
11      inicio
12          leia home) // referente ao botão com desenho de casinha.
13          se falso move para cima;
14          até verdadeiro bater na chave de fim de curso; // interrompe o
// movimento na origem do plano cartesiano.
15          aguardar entrada de dados;
16      fim
17 fim
```

Os algoritmos de movimentação dos eixos são relevantes pois permitem ao usuário a definição da velocidade e a quantidade que os eixos podem se movimentar em cada comando. Todos os eixos devem ser controlados cuidadosamente pelos usuários para evitar acidentes, desregulamento ou mesmo colisões com os objetos impressos e que estão em processo de cura.

```

1   algoritmo MOVIMENTO_XY

// O algoritmo de movimento nos eixos X e Y foi definido para permitir uma
// movimentação adequada, utilizando unidades em metros. O código é similar ao
// MOVIMENTO_Z.

2   variaveis velocidade, valores_movimento: real
3       multiplicador_distancia: inteiro
4   constante home: lógico

5   inicio // não começa com letra maiúscula e não é utilizado acento nos
// algoritmos.

6   escreva ('Insira a Velocidade Desejada');
7   leia (velocidade);

8   escreva ('velocidade=  m/min', velocidade) // velocidade em metros por
// minutos.

9   valores_movimento (0.01, 0.1, 1, 5) // unidades de medida em metros.

10  multiplicador_distancia (0.01 * 10) // multiplica a unidade básica de medida para
// aumentar a distância percorrida.

11  inicio

12  leia (home) // referente ao botão com desenho de casinha.

13  se falso move para o lado

14  até verdadeiro bater na chave de fim de curso

15  <aguarde entrada de dados>

```

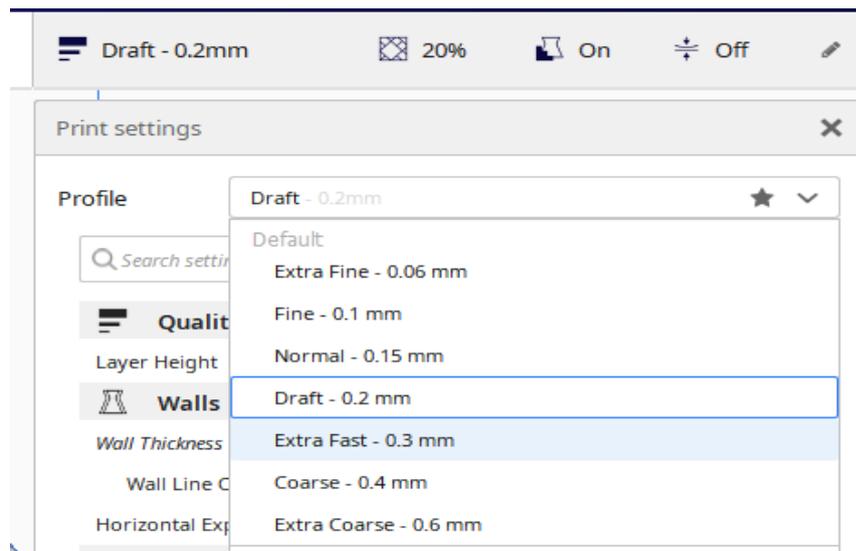
16 fim

17 fim

6.1.3 Bloco para definição de diâmetro do bico extrusor

As espessuras de camadas produzidas por impressoras tridimensionais de plástico variam entre 0,06 mm até 0,6 mm (cf. Figura 44). A espessura em medidas menores que o milímetro permite detalhamento e qualidade visual para as peças. Contudo, a espessura nanométrica das camadas cria impressões que levam horas ou até mesmo dias para serem concluídas.

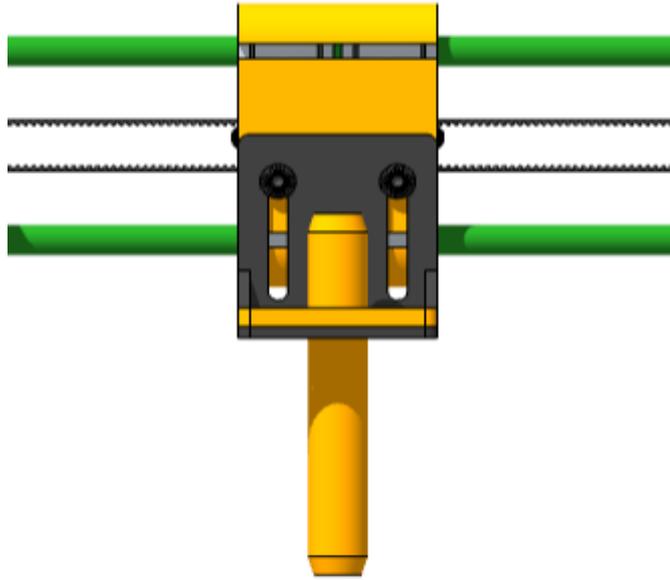
Figura 44 – Definições de espessuras de camadas do software proprietário *ULTIMAKER Cura*



Fonte: o autor.

Uma impressora de argamassa para produção de edificações em escala natural deve produzir camadas com espessuras que são medidas em centímetros (cm). A impressora produzida na FAU/UnB tem extrusor de bico fixo com abertura de 1 cm de diâmetro, o que possibilita camadas com espessura que podem variar de 0,3 cm até 1 cm. Uma impressora de tamanho natural com bico de extrusão de 5 cm de diâmetro pode produzir camadas de 1 cm a 5 cm de espessura.

Figura 45 – Bico de impressão fixo – 1 cm de diâmetro. Impressora produzida na FAU/UnB.



Fonte: Buzon, 2020.

Camadas estreitas aumentam a probabilidade de entupimento do sistema de extrusão devido à pressão necessária para expelir o material e produzem ondulações e marolas, pode ser visto na Figura 46:

Figura 46 – Camadas com ondulações e marolas. Impressora de barro, 3D Wasp, Itália



Fonte: Empresa 3D Wasp, 2021.

As camadas não podem ser mais altas que o diâmetro do bico de extrusão. Camadas muito grossas não permitem a fixação adequada da argamassa, de forma que compromete a qualidade visual das edificações e, principalmente, a sua integridade estrutural.

```
1   algoritmo BICO_EXTRUSAO

// O código de bico de extrusão tem por objetivo permitir a configuração do diâmetro
do bico utilizado em diferentes impressoras de argamassa. O algoritmo calcula a
espessura das camadas para manter a proporção referente ao diâmetro e apresenta
as possíveis espessuras de camadas mínimas e máximas.

2   variaveis diametro_bico: inteiro;

3       camada_min, camada_max: real;

4   inicio

5       (diametro_bico) > [0]; // diâmetro do bico em cm. O bico aparece como zero
para permitir que seja modificado pelo usuário.

6   escreva ('Insira o Diâmetro do Bico de extrusão');

7   leia (diametro_bico);

8   escreva ('Diâmetro do Bico=  cm', diametro_bico);

9       inicio

10      conecta ao bloco de fatiamento;

11      define ('diametro_bico', [5]); // por padrão, a ferramenta deve mostrar o
bico de extrusão com 5 centímetros de diâmetro.

12      se diametro_bico ← 5; // ← significa igual (=).

13      entao ('camada_min', [1]); // camada mínima de 1 centímetro.

14          ('camada_max, [5]); // camada mínima de 5 centímetros.

15          inicio

16      se diametro_bico > 5;

17          entao ('camada_min', [ (1*diametro_bico)/5 ]; // camada mínima
proporcional ao bico de extrusão para manter a proporção centímetro.

18          ('camada_max, [(5*diametro_bico)/5]) // camada mínima
proporcional ao bico de extrusão para manter a proporção centímetro.
```

19 **fim**
 20 **fim**
 21 **fim**

6.1.4 Definição de algoritmo de pausa e retomada de impressão

É necessário indicar pausas na impressão para que seja possível instalar tubulações, vigotas ou barras estruturais de aço. Outra necessidade apresentada pelo material é a de programar pausas para permitir o processo de cura da argamassa.

O programa completa o menu apresentado na Figura 47 – *Load file* (carregar arquivo), *SD* (*Secure Digital*), *Print* (Imprimir), *Pause* (Pausa) e *Off* (Desligado). O algoritmo proposto define que o usuário pode interromper a impressão para que seja inserido um complemento na edificação ou para garantir a cura da argamassa.

Figura 47 – Menu do programa *Printrun/Pronterface* para escolha de arquivos, início de impressão e pausa



Fonte: o autor.

O algoritmo apresenta:

- Processo de interrupção dividido em “**pausa**” – parada sem mover o extrusor do lugar;
- Pausa com movimentação que permite o extrusor ser movido automaticamente um metro para frente, para tirar o bico do local de impressão, e retoma o trabalho na posição original;
- Definição da “interrupção para processo de cura” com marcações no código G para retomada da impressão no mesmo ponto de interrupção. A

interrupção para processo de cura é necessária também para permitir que o sistema de extrusão seja limpo ao final da etapa, ou seja, os eixos precisam voltar para a origem dos espaços; e

- O algoritmo é definido para salvar a coordenada cartesiana do código G no momento da pausa para que a impressão seja retomada depois de horas ou dias de espera.

1 algoritmo PAUSA_RETOMADA

// É um dos códigos mais importantes do estudo. Foi escrito para permitir que os usuários possam interromper o movimento dos eixos e retomar a impressão sem mover o bico para pausas rápidas. Apresenta a possibilidade de pausas longas com movimentação dos eixos e definição de posição dos eixos para evitar possíveis colisões com os limites das impressoras.

// O código apresenta a pausa para cura. Processo de interrupção da impressão para que o material utilizado possa ganhar resistência. A pausa para cura salva a coordenada do código G para que a impressão possa ser retomada do exato ponto de parada.

2 variaveis pausa_mov, pausa_cura, salvar_prog, pausa: logico

3 // pausa_mov para a impressão e movimenta o eixo X para direita. pausa_cura movimenta os eixos para a origem dos eixos (*home*). O comando identifica a linha de código e salva o código G sem as linhas que já foram executadas.

4 coordenada: vetor [4] // número 4 é referente às posições dos elementos que compõem um vetor, variável com vários elementos. No caso, o vetor “coordenada” é referente à linha do código G que indica posição “G” referente ao código indicado para o movimento da máquina; X, Y e Z são referentes ao posicionamento do extrusor no espaço. Ex.: G1 X 100 Y 750 Z 300. Não é necessário comando para extrusão (E) e para velocidade (F).

5 constante diretorio: vetor [3] // encontrar pasta.

6 **inicio**

7 **escreva** ('PAUSA COM MOVIMENTO' =,pausa_mov)

```

8   se pausa_imp falso entao
9       executa < codigo G >
10  se pausa_imp verdadeiro entao
11      pausa motores
12      // x← + 100 // move eixo X um metro para direita. Obs: base impressora
13      // se 100 > X > 501 entao.
14      Y← + 100 // o algoritmo prioriza o movimento do eixo X por ser
15      // se Y > 801 entao.
16      Y← - 100 // eixo Y move na posição oposta para não colidir com
17      // As linhas 8, 9, 10 estão comentadas (//) pois se referem a impressoras de
18      // tamanho natural.
19  se 20 > X > 43 entao
20      X← + 20
21      senao
22      X← - 20
23  se 20 > Y > 69 entao
24      Y← + 20
25      senao
26      Y← - 20
27  // As linhas 12, 13, 14 referem-se à impressora da FAU/UnB.
28  fim
29  inicio

```

```

29  escreva ('Pausa para Cura' =,pausa_cura)
30  se pausa_cura falso entao
31      {execute < codigo G >}
32  se pausa_cura verdadeiro entao
33      pausa motores
34      leia coordenada < codigo G > // salva a linha de código G para que o
sistema retome a impressão depois da cura do material.
35      X = 0, Y = 0, Z = 0; // movimenta os eixos para a origem do plano
cartesiano.
36      escreva ('Retomo deste ponto = [coordenada]', coordenada);
37      escreva ('Salvar Progresso= ', salvar_prog')
38      se salvar_prog falso entao
39          leia coordenada
40      se salvar_prog verdadeiro entao
41          leia coordenada em diretorio = (C:) desktop>Gcode: CURA > //
buscar a linha de código G em um diretório na área de trabalho do computador.
42      leia < diretorio >
43  fim

```

6.2 Algoritmo de controle de velocidade

O programa indica movimentos de eixos sem expelir material. Esses movimentos são importantes para a impressão de partes diferentes da edificação, como vãos, aberturas ou pilares. O *Printrun/Pronterface* apresenta um conjunto de funcionalidades para a definição de velocidade de impressão, movimento dos eixos, e a definição de fluxo de impressão, como será visto na Figura 48.

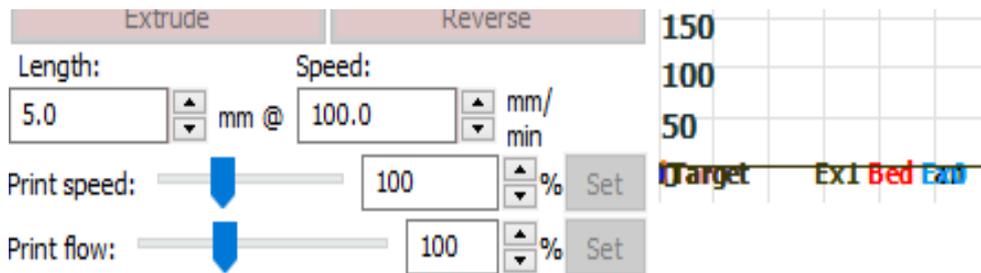
6.2.1 Bloco de algoritmo para controle de velocidade de movimento dos eixos sem extrusão

A ferramenta de definição de velocidade nativa do *Printrun/Pronterface* apresenta duas funcionalidades:

- **Print speed** (velocidade de impressão): modifica a impressão aumentando ou diminuindo a velocidade de movimentação dos eixos; e
- **Print Flow** (fluxo de impressão): permite aumentar ou diminuir a quantidade de material que é expelido pela máquina durante o processo de impressão.

Os parâmetros indicados no algoritmo permitem que o usuário ajuste a impressão para uma determinada argamassa ou traço.

Figura 48 – Ícones de controle de velocidade para impressoras *desktop* para impressão em plástico



Fonte: o autor.

A velocidade padrão da ferramenta é apresentada em milímetros por minuto (mm/min) e o fluxo de material em milímetros cúbicos. O algoritmo de velocidade é definido para apresentar a velocidade de movimento em metros por minuto (m/min) para ser compatível com o que é verificado no visualizador 3D, quadrícula de base de impressão. O algoritmo de fluxo deve ser apresentado em centímetros para ser compatível com a medida de espessura de camadas.

Contudo, não se faz necessário modificar a programação de movimentação. O algoritmo deve apresentar a conversão para ser compatível com a velocidade de movimento de milímetros por minuto para metros por minuto.

```
1 algoritmo VELOCIDADE_SEM_EXTRUSAO
2 // O bloco de velocidade sem extrusão permite que os eixos se movimentem
entre os pontos de impressão. Esses pontos podem variar para definir vãos ou
diferenças de alturas dos elementos arquitetônicos.
// A velocidade é definida utilizando como parâmetros a plasticidade da argamassa. O
usuário insere os limites de liquidez e plasticidade e o algoritmo apresenta velocidades
adequadas para a movimentação. Esse algoritmo é definido com a plasticidade para
evitar que a argamassa escorra pelo bico durante o processo de movimentação dos
eixos.
3 variaveis coordenada1, coordenada2, var_tempo, vel_semE: real; // definida
em metros por minuto (m/min).
4 plasticidade, LL, LP: real; // LL = Limite de Liquidez e LP = Limite de
Plasticidade.
5
6 inicio função vel_semE;
7 leia (coordenada1);
8 leia (coordenada2);
9 vel_semE ← (coordenada1 - coordenada2)/ var_tempo; // definição de
velocidade de movimento.
10 retorne vel_semE;
11 fimfunção
12
13 inicio
14 escreva ('Qual é o Limite de Liquidez= ', LL);
15 leia (LL); // o usuário insere o Limite de Liquidez.
16 escreva ('Qual é o Limite de Plasticidade= ', LP);
```

```

17  leia (LP) // o usuário insere o Limite de Plasticidade.
18  plasticidade ← (LL – LP) // o índice de plasticidade é definido pela diferença
entre o Limite de Liquidez (LL) e o Limite de Plasticidade (LP).
19  // É sugerido escolha de argamassas com índice de plasticidade de 4% a 11%
(MATUK e HAY apud NEVES; FARIA, 2011).
20      inicio
21  se ( 7 >= plasticidade >= 11 );
22      entao vel_semE = 3 ou vel_semE = 4 vel_semE = 5 // se a plasticidade
da argamassa for maior ou igual a 7% e menor ou igual a 11%, permite a inserção de
três velocidades predefinidas.
23  // FAU/UnB e dimensões reduzidas usar parâmetros reduzidos em 100%.
24  // Linhas comentadas para serem substituídas para o uso da impressora de
menor dimensão da FAU/UnB.
25      // entao vel_semE = 0.3. ou vel_semE = 0.4 vel_semE = 0.5.
26      escreva ('Velocidades permitidas 3 ou 4 ou 5 m/min = ' vel_semE); //
permite que o usuário insira a velocidade desejada.
27      // escreva ('Velocidades permitidas 0.3 ou 0.4 ou 0.5 m/min = '
vel_semE).
28      enquanto vel_semE <> 3 ou 4 ou 5 faça
29      // enquanto vel_semE <> 0.3 ou 0.4 ou 0.5 faça.
30      leia ( vel_semE);
31      senao
32      se ( 4 >= plasticidade >= 6.9 );
33      entao vel_semE = 1 ou vel_semE = 2 vel_semE = 2.90 // se a
plasticidade da argamassa for maior ou igual a 4% e menor ou igual a 6,9%, permite
a inserção de três velocidades predefinidas.
34  // Impressora FAU/UnB com dimensões reduzidas. Os parâmetros são
reduzidos em 100%.

```

```
35 // Linhas comentadas para serem substituídas para o uso da impressora de
    menor dimensão da FAU/UnB.

36 // entao vel_semE = 0.1 ou vel_semE = 0.2 vel_semE = 0.29.

37 escreva ('Velocidades permitidas 1 ou 2 ou 2.90 m/min = ' vel_semE);
// permite que o usuário insira a velocidade desejada.

38 // escreva ('Velocidades permitidas 0.1 ou 0.2 ou 0.29 m/min =
'vel_semE);

39 enquanto vel_semE <> 1 ou 2 ou 3 faça

40 // enquanto vel_semE <> 0.1 ou 0.2 ou 0.3 faça.

41 leia ( vel_semE);

42 fim

43 fimse

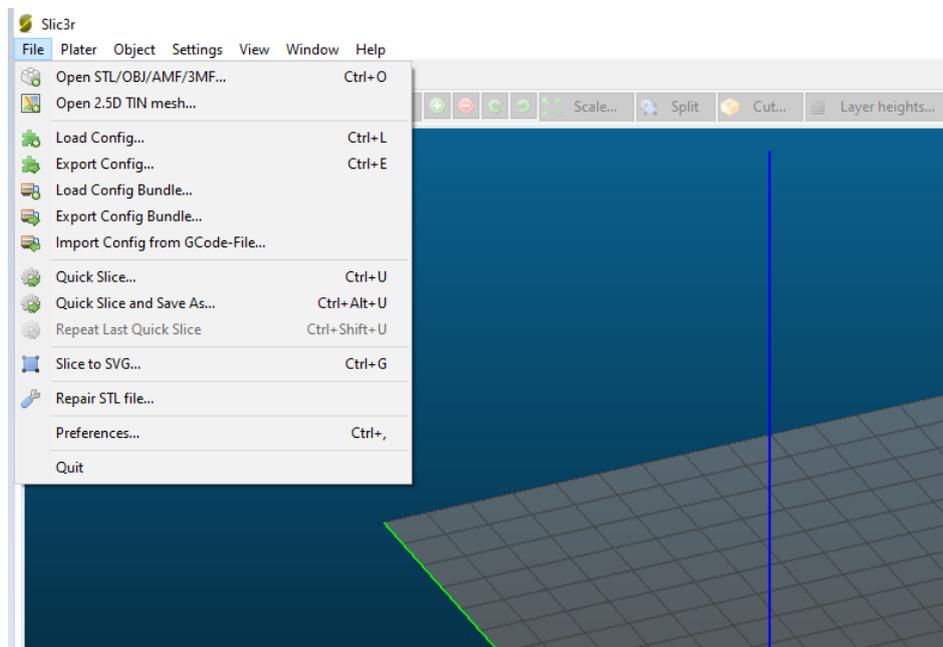
44 fimse

45 fim
```

7 Programa *Slic3r* – requisitos de especificação de *software*

O programa *Slic3r* é um programa gerador de percursos para impressoras 3D (*toolpath generator*). O *software* consegue ler diversos tipos de arquivos destinados à produção de modelos para impressão 3D (.STL, .OBJ, AMF, 3MF). A ferramenta possui a função de reparar arquivos .stl, importante para usuários de impressoras 3D (cf. Figura 49). O principal objetivo de um programa gerador de percursos é converter os modelos virtuais em instruções do tipo código G.

Figura 49 – Menu file. Programa *Slic3r*



Fonte: o autor.

O *Slic3r* nasceu em 2011 na comunidade RepRap, responsável por desenvolver, construir e distribuir de forma gratuita as informações referentes à construção de ferramentas para impressão 3D. O programa permite a implementação de novas ideias e experimentações.

O *Slic3r* é um programa de código aberto e não possui relação comercial com nenhuma empresa ou fabricante de impressoras tridimensionais. É uma plataforma

compatível com diversos tipos de linguagens para código G, incluindo o firmware⁵ “Marlin” utilizada na impressora da FAU/UnB (cf. Figura 50).

Figura 50 – Licença de código aberto para uso do *Firmware Marlin* para impressoras 3D



```

PASTER_Ub_BUSON.RENAN_rev.29.12 | Arduino 1.8.16 (Windows Store 1.8.51.0)
Arquivo Editar Sketch Ferramentas Ajuda
Carregar
Conditionals.h Conditionals_LCD.h Conditionals.h

/**
 * Marlin 3D Printer Firmware
 * Copyright (C) 2016 MarlinFirmware [https://github.com/MarlinFirmware/Marlin]
 *
 * Based on Sprinter and grbl.
 * Copyright (C) 2011 Camiel Gubbels / Erik van der Zalm
 *
 * This program is free software: you can redistribute it and/or modify
 * it under the terms of the GNU General Public License as published by
 * the Free Software Foundation, either version 3 of the License, or
 * (at your option) any later version.
 *
 * This program is distributed in the hope that it will be useful,
 * but WITHOUT ANY WARRANTY; without even the implied warranty of
 * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
 * GNU General Public License for more details.
 *
 * You should have received a copy of the GNU General Public License
 * along with this program. If not, see <http://www.gnu.org/licenses/>.
 */

/**
 * About Marlin
 *
 * This firmware is a mashup between Sprinter and grbl.
 * - https://github.com/kliment/Sprinter
 * - https://github.com/simen/grbl/tree

```

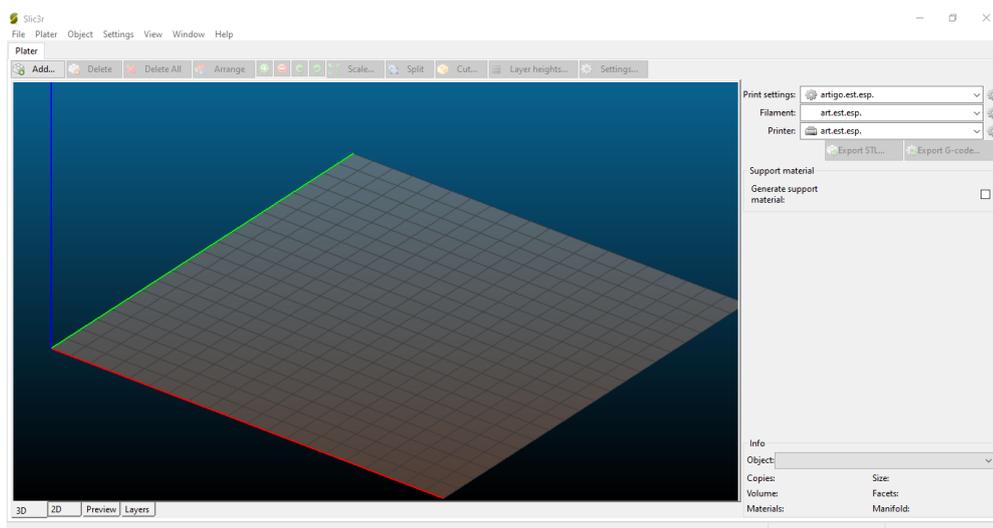
Fonte: o autor.

O programa é desenvolvido com a linguagem C++ (C mais mais). O C++ é uma linguagem de alto nível e tem origem na linguagem C. É uma programação simbólica, ou seja, utiliza símbolos no lugar de palavras por extenso (DAVIS, 2011).

⁵ *Firmware* é uma classe específica de programas de computador que fornece controle de baixo nível para o hardware específico. A impressora utiliza um firmware específico para a controladora Arduino mega 2560.

A ferramenta pode ser utilizada sozinha ou como um *plugin* (complemento em outros *softwares*) como fatiador para programas de impressão tridimensional. A versão independente, demonstrada pela Figura 51, permite configurar as características da impressora utilizada e possui um visualizador 3D que simula a base de impressão.

Figura 51 – Software independente *Slic3r*



Fonte: o autor.

Os códigos referentes ao programa estão disponíveis na internet, através do *site GitHub.com*, e permite a modificação dos blocos de códigos para adaptar seu funcionamento para o uso proposto em uma impressora tridimensional em escala 1:1.

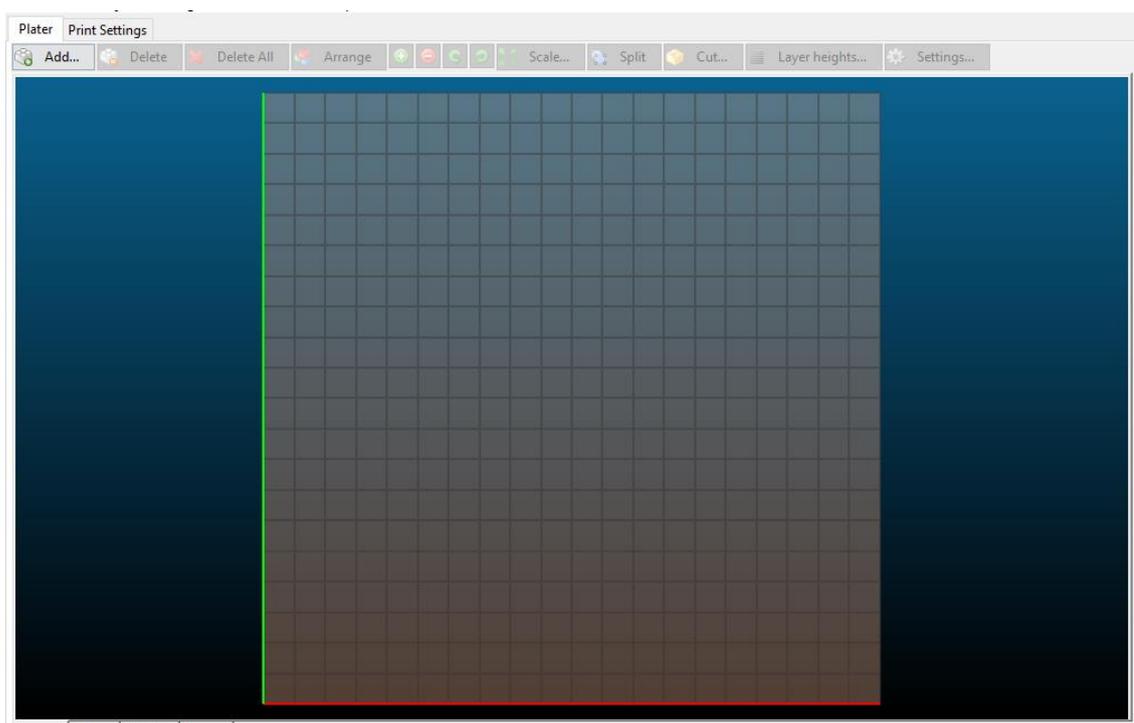
O uso do programa e a leitura dos códigos em C++ permite que usuários possam identificar os diferentes blocos de código desenvolvidos para o funcionamento da ferramenta digital. Contudo, ler o código não é uma tarefa de fácil entendimento. É necessário ter um conhecimento de lógica de programação e sintaxe da língua definida para o programa, pois cada linguagem possui sistemas e códigos únicos para o seu funcionamento.

O código G é mais simples de ser entendido por possuir uma série de códigos predefinidos para o funcionamento das máquinas CNC.

7.1 Algoritmo para volume de impressão em metros

A base de impressão do programa é apresentada em centímetros (cf. Figura 52), o que dificulta a visualização do modelo virtual em tamanho natural. A unidade de medida deve ser modificada de centímetros para metros para permitir a compatibilização com a ferramenta *Printrun/Pronterface*.

Figura 52 – Vista superior da base de impressão do programa *Slic3r*



Fonte: o autor.

1 algoritmo VOLUME_METROS

// O código apresenta a definição de valores relativos aos eixos X, Y e Z em metros e subdivididos em centímetros. O algoritmo foi escrito para permitir que o usuário modifique os espaços de movimentação dos eixos. Contudo, foram definidos limites e o algoritmo avisa ao usuário se os dados inseridos são adequados ou não e só continua o funcionamento se os dados forem inseridos de forma corretos.

2 variáveis unidade X, unidade Y, unidade Z: inteiros;

3 // grade←(1 cm; 10 cm); // definição de grade dividida em centímetros para impressora FAU/UnB de medidas reduzidas.

4 // subgrade←(10 mm por 10 mm) // a base deve apresentar subdivisões em milímetros para impressora FAU/UnB de medidas reduzidas.

5 grade←(10 cm; 100 cm); // definição de grade dividida em centímetros/metros para utilização em impressoras de tamanho natural.

6 subgrade←(100 cm por 100 cm) // a base deve apresentar subdivisões em centímetros para facilitar a organização da impressão.

7 dimensao_volume: inteiros

8 **inicio**

9 **escreva** ('Qual é a medida da base em X= ', unidade X);

10 **leia** (unidade X);

11 **se enquanto** unidade X > 600 **faça**

12 // **se enquanto** unidade X > 63 **faça** // medida de X para a impressora reduzida da FAU/UnB. Linha comentada para não influenciar o funcionamento do algoritmo.

13 **escreva** ('Medida inválida. A impressora não pode ter mais de 6 metros de largura');

14 // **escreva** ('Medida inválida. A impressora não pode ter mais de 63 centímetros de largura'); // impressora reduzida da FAU/UnB.

15 **escreva** ('Qual é a medida da base em Y' = unidade Y);

16 **leia** (unidade Y);

17 **se enquanto** unidade Y > 900 **faça**

18 // **se enquanto** unidade Y > 89 **faça** // impressora reduzida da FAU/UnB.

19 **escreva** ('Medida inválida. A impressora não pode ter mais de 9 metros de comprimento');

20 // **escreva** ('Medida inválida. A impressora não pode ter mais de 89 centímetros de comprimento'); // impressora reduzida da FAU/UnB.

21 **escreva** ('Qual é a medida da base em Z' = unidade Z);

```

22  leia (unidade Z);
23  se enquanto unidade Z > 400 faça
24  // se enquanto unidade Z > 41,5 faça // impressora reduzida da FAU/UnB.
25  // escreva ('Medida inválida. A impressora não pode ter mais de 41,5
centímetros de altura');
26  escreva ('Medida inválida. A impressora não pode ter mais de 4 metros de
altura');
27  dimensao_volume← (unidade X, unidade Y, unidade Z)
28  leia (unidade X, unidade Y, unidade Z, 0, 0, 0);
29  escreva ('Base = "unidade X", 'por', "unidade Y", unidade X, Unidade Y');
30  escreva ('Altura Máxima para impressão = ', unidade Z);
31
                                     fim
32
                                     fimse
33
                                     fimse
34
                                     fimse
35  fim

```

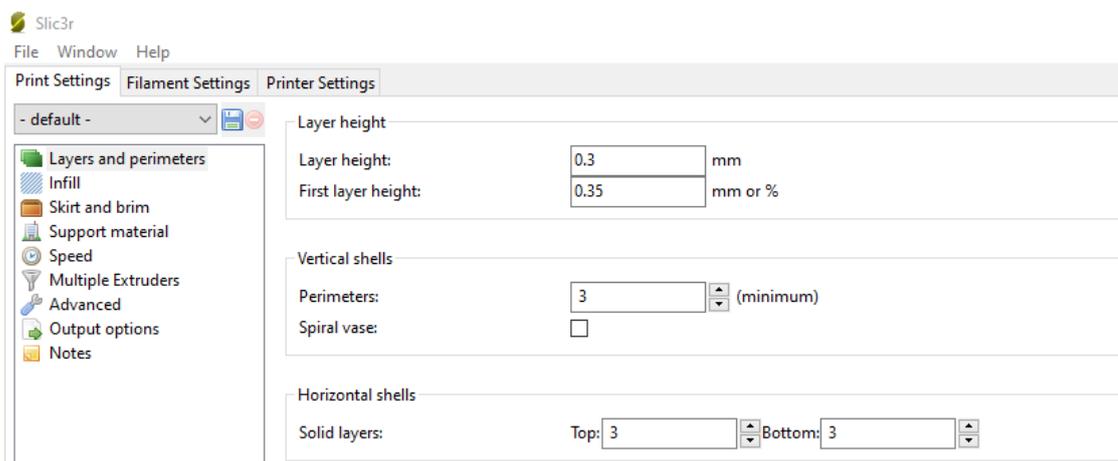
7.2 Algoritmo relacionado a retração da argamassa para compensar possível deformação das camadas

É necessário considerar o abatimento das argamassas de cimento e terra. As impressoras 3D de extrusão de material fundido (FDM) não calculam o abatimento do material, pois o filamento de material plástico sofre o processo de resfriamento em poucos segundos, o que garante a integridade da impressão.

O abatimento deve ser um parâmetro inserido no programa de fatiamento para compensar a diferença entre o modelo virtual e o modelo impresso. O algoritmo define a indicação de mais camadas de acordo com a relação de abatimento do material

inserido pelo usuário. A Figura 53 apresenta o menu do programa de código aberto *Slic3r* que permite a definição dos parâmetros de altura das camadas. O algoritmo apresenta a possibilidade de inserir no menu *Layer height* (altura de camada) o abatimento da argamassa utilizada.

Figura 53 – Menu para definição de alturas de camadas



Fonte: o autor..

1 algoritmo RETRAÇÃO

// O parâmetro de retração foi considerado importante para o estudo. Algumas argamassas e traços podem apresentar essa característica e variação de dimensões.

// O código permite que o usuário insira o limite de retração e apresente por meio de mensagens se a argamassa pode apresentar problemas como rachaduras. O programa utiliza os dados relacionados ao bico de extrusão e espessuras de camadas e compensa automaticamente o efeito gerado pela retração.

2 **inclua** biblioteca Matematica←mat; // permite arredondar resultados diminuindo as casas decimais.

3 variaveis LR, retracao, camada, altura_objeto, camada_final: inteiros; // LR – significa limite de retração.

4 // A altura do objeto é referente à altura do elemento arquitetônico, representado de forma virtual. Exemplos: paredes, pilares, bancos.

5 altura_retracao, quantidade_camadas: real

6 **inicio**

```
7   escreva ('Qual é o Limite de retração= ', LR);
8   leia (LR); // o usuário insere o limite de retração aferido.
9       retracao ← LR /100;
10  se LR <= 2/100 entao
11      escreva ('Altura de camada= cm', camada); // usuário insere a altura de
camada desejada.
12  se LR > 2/100 entao
13      escreva ('É possível que sua impressão apresente rachaduras'); // caso
os corpos de prova não apresentem retração maior que 2 centímetros e nem
tampouco trincas, as argamassas de terra podem ser utilizadas para componentes e
elementos construtivos.
14      leia (camada);
15      inicio
16          busca BICO_EXTRUSAO // resgata bloco de algoritmo definido
para bico de extrusão e, conseqüentemente, define as alturas mínimas e máximas
para as alturas de camada.
17          altura_retracao ← altura_objeto/ (1.00+(LR/100));
18      // exemplo: 300*1.02 = 294 (O resultado da conta é 294,117. Contudo, a
variável altura_retracao foi definida como real, o que faz com que as linguagens de
computador suprimam as casas decimais).
19      quantidade_camadas ← ((altura_objeto+(altura_abjeto-altura_retracao))
/camada))+1;
20      // exemplo: quantidade_camadas= (300 +(300-294)) /5 = 61+1 = 62
camadas.
21      // o resultado da conta (300 +(300-294)) /5 = 61,2. Contudo, a variável
quantidade de camadas foi definida como real, ou seja, não reconhece as casas
decimais. O número 1 serve para arredondar para cima, não importa qual é a divisão
apresentada.
```

22 // a soma de “1” é denominado “incrementar”, ou seja, é a soma de um valor constante que normalmente é o número 1 (FORBELLONE; EBERSPACHER, 2005).

23 **camada_final**←mat.(altura_objeto+(altura_abjeto–altura_retracao))/
quantidade_camadas, 3); // exemplo: 306/ 62 = 4,935.

24 // a parte final do comando (,3) é vinculado ao uso da biblioteca mat. e serve para indicar 3 casas decimais depois da vírgula.

25 **escreva** ('Quantidade de camadas= ', quantidade_camadas); // apresenta a quantidade de camadas. Exemplo: 62.

26 **escreva** ('Espessura da camada= cm', camada_final); // apresenta a espessura da camada corrigida. Exemplo: 4,935 cm.

27 **fim**

28 **fimse**

29 **fimse**

30 **fim**

31 **fim**

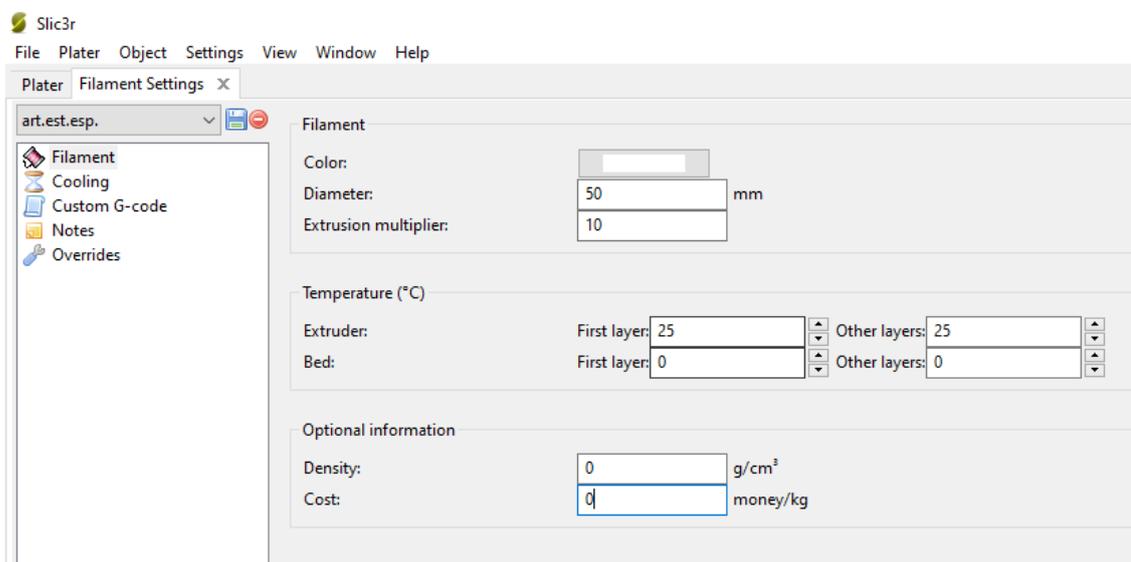
7.3 Algoritmo relacionado à plasticidade das argamassas para definir a velocidade de extrusão

O algoritmo possibilita a inserção de parâmetros de plasticidade e fluidez definidos pelo usuário. A plasticidade e a fluidez podem influenciar a velocidade dos eixos. Quanto mais fluida a argamassa, maior é a velocidade de movimento dos eixos sem extrusão para evitar que o material escorra pelo bico de extrusão. O algoritmo estabelece uma velocidade limite para evitar vibrações indesejadas e reposicionamento do extrusor fora do local de impressão definido pelo código G.

As impressoras 3D de mesa utilizam filamentos plásticos com diâmetros de 1,75 mm ou 3 mm de espessura. O programa *Slic3r* permite configurar o diâmetro do

material de extrusão utilizado na impressora e a temperatura necessária para fundir o material, como é possível ver na Figura 54:

Figura 54 – Menu para definição de diâmetro de filamento em impressoras FDM



Fonte: Autor.

A definição de diâmetro do filamento é uma característica do programa que pode ser mantida. É necessário mudar a unidade de medida do diâmetro de milímetros para centímetros.

1 algoritmo PLASTICIDADE_VELOCIDADE

2 // Neste algoritmo, são apresentadas linhas utilizadas em outros algoritmos pois são necessárias para definir parâmetros em partes diferentes do funcionamento da ferramenta.

// O algoritmo é similar ao criado para velocidade sem extrusão. O usuário pode inserir os dados e as características das argamassas e o código realiza os cálculos referentes à plasticidade. O programa usa como base as velocidades utilizadas no algoritmo VELOCIDADE_SEM_EXTRUSAO e, dependendo do índice, calcula uma compensação da velocidade com o objetivo de garantir a movimentação adequada dos eixos e a qualidade da impressão.

3 variaveis diametro_bico, vel_imp: inteiro; // vel_imp = velocidade de impressão dos modelos ou artefatos arquitetônicos.

```

4   plasticidade, LL, LP: reall; // LL = Limite de Liquidez e LP = Limite de
    Plasticidade

5   inicio

6   (diametro_bico) > [0]; // diâmetro do bico em cm. O bico aparece como zero
    para permitir que seja modificado pelo usuário

7   escreva ('Diâmetro do Bico= cm', diametro_bico);

8   leia ( diametro_bico);

9   escreva ('diametro_bico', [5]); // por padrão, a ferramenta deve mostrar o bico
    de extrusão com 5 centímetros de diâmetro.

10  inicio

11  escreva ('Qual é o Limite de Liquidez= ', LL);

12  leia (LL); // o usuário insere o Limite de Liquidez.

13  escreva ('Qual é o Limite de Plasticidade= ', LP);

14  leia (LP); // o usuário insere o Limite de Plasticidade.

15  plasticidade ← (LL – LP); // o índice de plasticidade é definido pela
    diferença entre o Limite de Liquidez (LL) e o Limite de Plasticidade (LP).

16  iniciose

17  se ( 7>= plasticidade >=11 ) e (3 < vel_imp <5) entao

18  vel_imp ← vel_semE / 1.10 // a velocidade de impressão é igual à
    velocidade sem impressão reduzida em 10%. Esse parâmetro é automatizado para
    diminuir a quantidade de dados inseridos no sistema. A velocidade sem extrusão é
    utilizada como base por ser mais rápida e apresentar possíveis vibrações na estrutura.
    A velocidade sem extrusão foi definida no algoritmo
    VELOCIDADE_SEM_EXTRUSAO.

19  senao

20  se ( 4>=plasticidade >= 6.9 ) entao

```

21 `vel_imp ← vel_semE // a velocidade sem extrusão é mais lenta,`
permitindo que continue como o tipo de movimento, ou seja, a velocidade de impressão
é igual à velocidade sem extrusão.

22 `// não é necessário modificar esses parâmetros para impressão em escala`
natural ou para a impressora da FAU/UnB.

23 `fim`

24 `fimse`

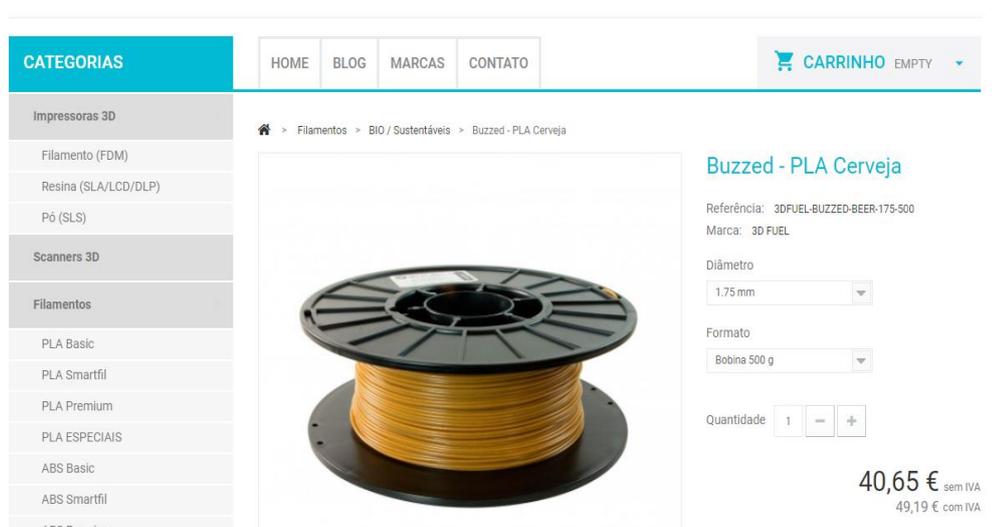
25 `fimse`

26 `fim`

7.4 Requisitos para modificação dos parâmetros de temperatura

A temperatura é um aspecto importante nas impressoras de material fundido, pois existem diferentes tipos de plásticos para impressão 3D. Os materiais têm propriedades diferentes devido à sua composição. Os plásticos podem ser produzidos com álcool, petróleo ou mesmo com materiais reciclados, como os subprodutos da cerveja, como o filamento *buzzed* (cf. Figura 55). Os diferentes tipos de plásticos possuem diferentes pontos de fusão para impressão. Plásticos de Acrilonitrila Butadieno estireno (*Acrylonitrile Butadiene Styrene* ou ABS) têm a temperatura de impressão entre 230°C e 250°C. O plástico de Ácido Polilático (PLA) tem sua temperatura de impressão entre 180°C a 200°C.

Figura 55 – Filamento Buzzed. Produzido com resíduos da fabricação de cerveja



Fonte: Filament2print, 2021.

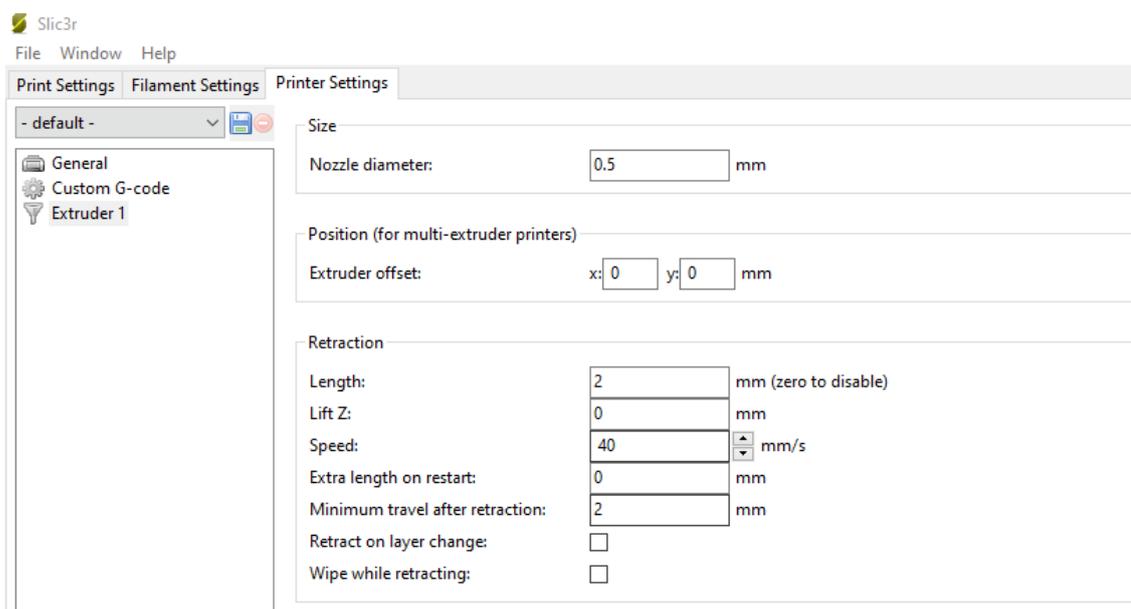
Os programas de impressão com plástico possuem *menus* para calibragem de retração do material durante o processo de impressão, conforme será demonstrado pela Figura 56. A retração visa garantir uma boa qualidade.

O parâmetro retração, em impressoras do tipo *desktop*, faz com que a impressora puxe o plástico no sentido oposto ao de extrusão para evitar que a alta temperatura do bico de impressão (entre 180°C e 250°C) faça o material plástico escorrer sobre as camadas.

O parâmetro relacionado ao aquecimento do plástico é apresentado no algoritmo como o que deve ser substituído pela plasticidade da argamassa, que não precisam ser aquecidas para permitir o processo de impressão. O parâmetro relacionado à plasticidade deve ser ligado aos de velocidade de movimento dos eixos. Quanto mais fluida uma massa, mais rápido será o movimento dos eixos da impressora.

O parâmetro vinculado à plasticidade foi integrado ao Algoritmo PLASTICIDADE_VELOCIDADE e algoritmo VELOCIDADE_SEM_EXTRUSAO.

Figura 56 – *Slic3r*. Menu de parâmetros de retração



Fonte: o autor.

O algoritmo de fatiamento direcionado para projeto de edificação com impressão em argamassa não apresenta o sistema de retração. O bloco de código das ferramentas digitais utilizadas no estudo devem ter os blocos relacionados a retração comentados⁶ ou ter seus parâmetros modificados para valores mínimos para não alterar o funcionamento do *software*.

7.5 Bloco de algoritmo para definição dos perímetros das linhas de impressão vinculado ao tamanho do bico extrusor

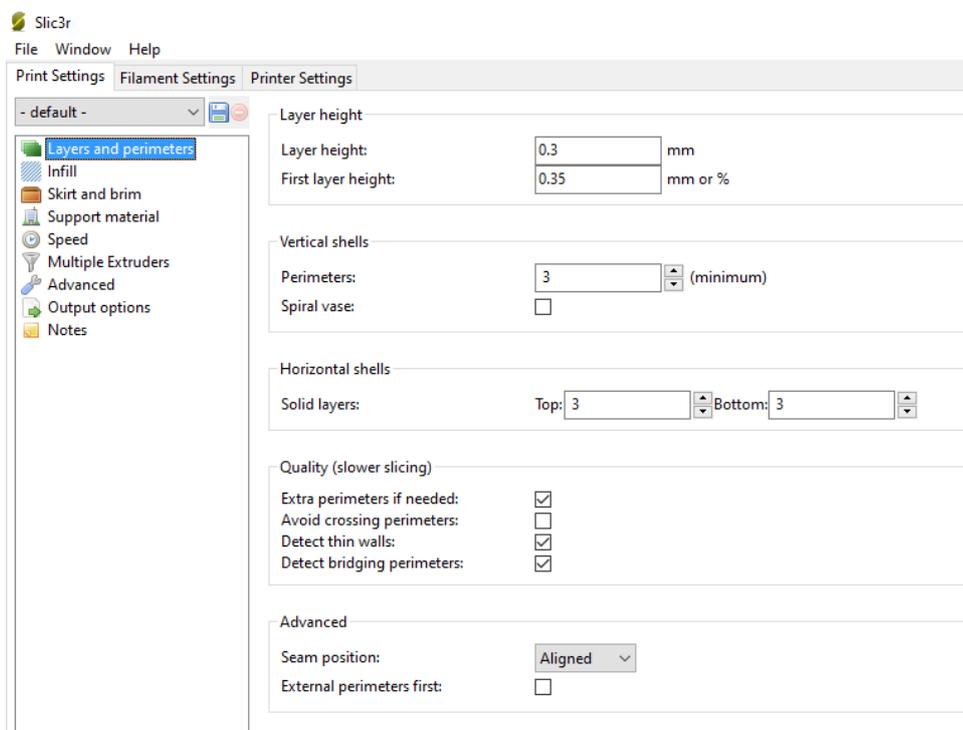
O diâmetro do bico de extrusão e a espessura definida para a altura da camada influencia a largura da linha de argamassa.

O *Menu Print settings* (Configurações de impressão, descrito pela Figura 57) do programa *Slic3r* disponibiliza vários parâmetros para configuração da impressão e, conseqüentemente, do modelo a ser impresso. As possibilidades de configuração são

⁶ Linhas de código comentadas são trechos do código que o programa não utiliza no seu funcionamento. São utilizadas para indicar orientações para programadores ou para desligar linhas de código que não estão sendo usadas em uma versão específica do código, mas que pode ser útil em uma outra versão (MILETTO; NICOLAO; OKUYAMA, 2014).

destinadas a impressão FDM, mas podem ser modificadas para impressão com outros materiais.

Figura 57 – *Slic3r*: Configurações de impressão

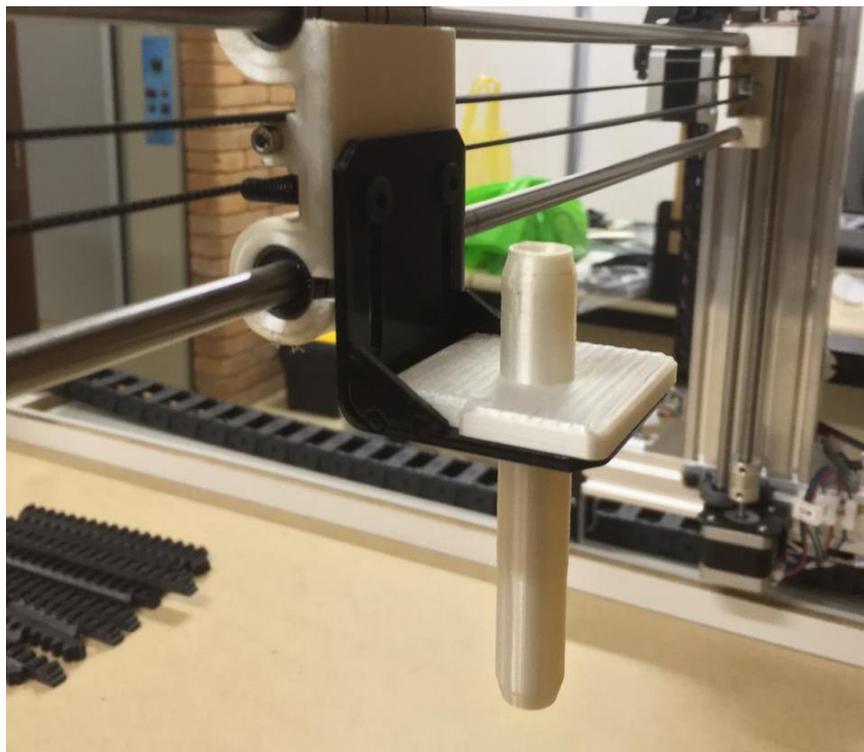


Fonte: o autor.

7.5.1 Layer height (altura de camada)

O modelo de impressora 3D proposto e construído para dar apoio ao estudo possui o bico de extrusão fixo e com formato circular (*vide* Figura 58). Portanto, a altura da camada modifica a largura final esperada para as linhas de camada. O programa *Slic3r* apresenta as alturas de camadas das impressoras de plástico em milímetros. O algoritmo define a mudança da unidade de medida das alturas de camadas de milímetros para centímetros para facilitar o raciocínio do usuário e manter a compatibilidade com os demais parâmetros.

Figura 58 – Bico de extrusão feito com impressora 3D FDM para impressão de argamassa



Fonte: o autor.

É necessário mudar a indicação de unidade de medida no código do programa. O comando “altura de camada” não necessita de parâmetros predefinidos, pois o usuário deve ter a liberdade de escolher qual é a altura de camada adequada para a edificação proposta.

O parâmetro para altura das camadas foi definido no algoritmo BICO_EXTRUSAO. Contudo, é necessário definir um algoritmo para a primeira camada. A camada inicial tem como função permitir uma boa aderência da argamassa ao solo e sem falhas para que as demais apresentem a qualidade adequada, sem falhas e sem ondulações.

1 algoritmo PRIMEIRA_CAMADA

// A camada inicial é calculada por último, pois utiliza a definição de retração e, conseqüentemente, a espessura das demais camadas do objeto. A primeira camada sofre uma diminuição de 30% para ajudar na aderência à base de impressão. O objetivo é garantir uma camada inicial homogênea, o que contribui para a qualidade das próximas camadas.

2 variaveis diametro_bico, camada_final, primeira_camada: inteiro;

3 **inicio**

4 **busca** BICO_EXTRUSAO; // utiliza o algoritmo que foi definido para o bico de extrusão.

5 **busca** RETRAÇAO; // utiliza o algoritmo que foi definido para o bico de retração.

6 primeira_camada←camada_final/1.3;// a primeira camada é aproximadamente 30% menor que as demais camadas que são definidas pela variável camada_final.

7 // Exemplo: $306 / 62 = 4,935 / 1.3 = 3,796$ cm de espessura.

8 **fimbusca**

9 **fimbusca**

10 **fim**

7.5.2 Vertical shells (perímetros verticais)

Os perímetros verticais são as cascas que delimitam os elementos construtivos relacionados a forma, limites das paredes e outros objetos. O sistema de definição de quantidades de perímetros permite definir se os elementos vão ser finos, grossos ou totalmente preenchidos (maciços). Não é necessário definir modificações no funcionamento do código fonte, pois o usuário define apenas a quantidade de

perímetros que acredita ser interessante para a execução do projeto. Veja-se a Figura 59:

Figura 59 - Configuração de impressão: *Vertical Shells* (perímetros verticais)

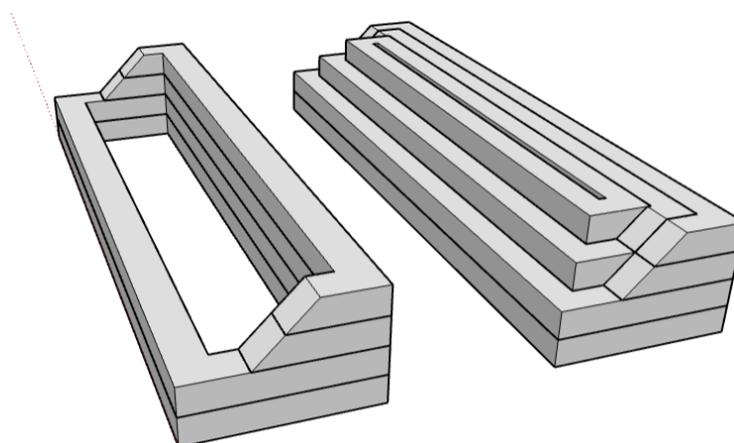
Vertical shells	
Perimeters:	3 (minimum)
Spiral vase:	<input type="checkbox"/>

Fonte: o autor.

É necessário redefinir os requisitos referentes às medidas. A ferramenta utiliza medidas em milímetros como padrão e deve ser modificada para centímetros. O usuário deve ter a liberdade de definir com quantas linhas de perímetros os elementos devem ser impressos. O parâmetro relacionado aos perímetros impacta diretamente no tempo de impressão, na quantidade de material utilizado e na capacidade estrutural das peças impressas.

A Figura 60 apresenta uma ilustração de como pode ser uma parede oca executada com apenas uma linha de perímetro e uma parede totalmente preenchida apenas com perímetros com 4 centímetros de largura de linha:

Figura 60 – Exemplo de parede sem preenchimento e totalmente preenchido com perímetros



Fonte: o autor.

7.5.3 Horizontal shells (preenchimentos horizontais)

O *menu horizontal shells* (preenchimentos horizontais) define quantas camadas devem ser feitas com 100% de preenchimento no início e no fim da impressão. É um

comando que não precisa ser modificado, permitindo, assim, que o usuário indique quantas camadas devem ter 100% de argamassa para fechar o volume impresso. Essa configuração pode ser vista na Figura 61:

Figura 61 – Configuração de impressão: *Horizontal Shells* (preenchimentos horizontais)

Fonte: o autor.

7.5.4 Comandos *Quality (slower slicing)* – *Qualidade (fatiamento lento)* e *Advanced (avançado)*

Os demais comandos relacionados ao *menu* “camadas e perímetros” são ajustes finos que não serão utilizados na definição dos demais comandos podendo ser comentados, ou seja, desabilitados, conforme Figura 62:

Figura 62 – *Slic3r. Quality (slower slicing)* – *Qualidade (fatiamento lento)* e *Advanced (avançado)*

Fonte: o autor.

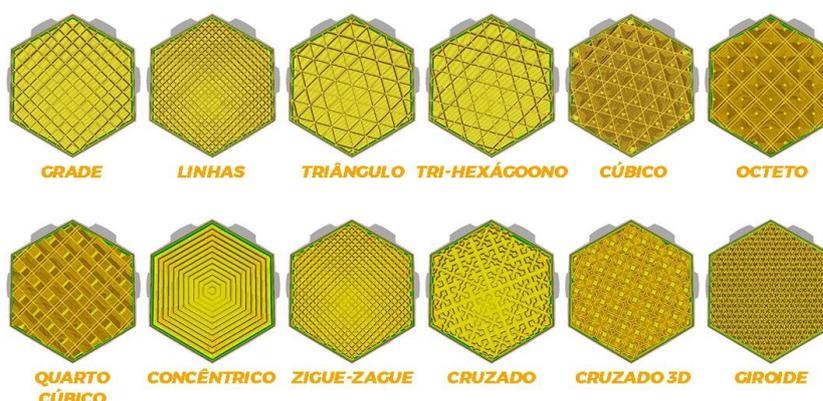
O estudo indica diversas linhas de código que devem ser comentadas, pois é importante manter as linhas de código da ferramenta para que outros desenvolvedores e pesquisadores, principalmente do campo da arquitetura e construção civil, possam modificar ou implementar novos parâmetros ao sistema. Um dos diferenciais da impressão 3D em grandes volumes, tamanho natural, é a pesquisa com diferentes tipos de materiais ou traços de argamassa. As ferramentas digitais

precisam ser versáteis e permitir a sua modificação para que novas pesquisas sejam incentivadas.

7.5.5 Comandos Infill (preenchimento)

Os elementos arquitetônicos impressos podem ser ocos ou com padrões de preenchimento⁷. Os padrões internos permitem uma melhor estabilidade e estruturação das peças impressas, cf. Figura 63:

Figura 63 – Exemplos de padrões de preenchimentos para impressoras tipo *desktop*

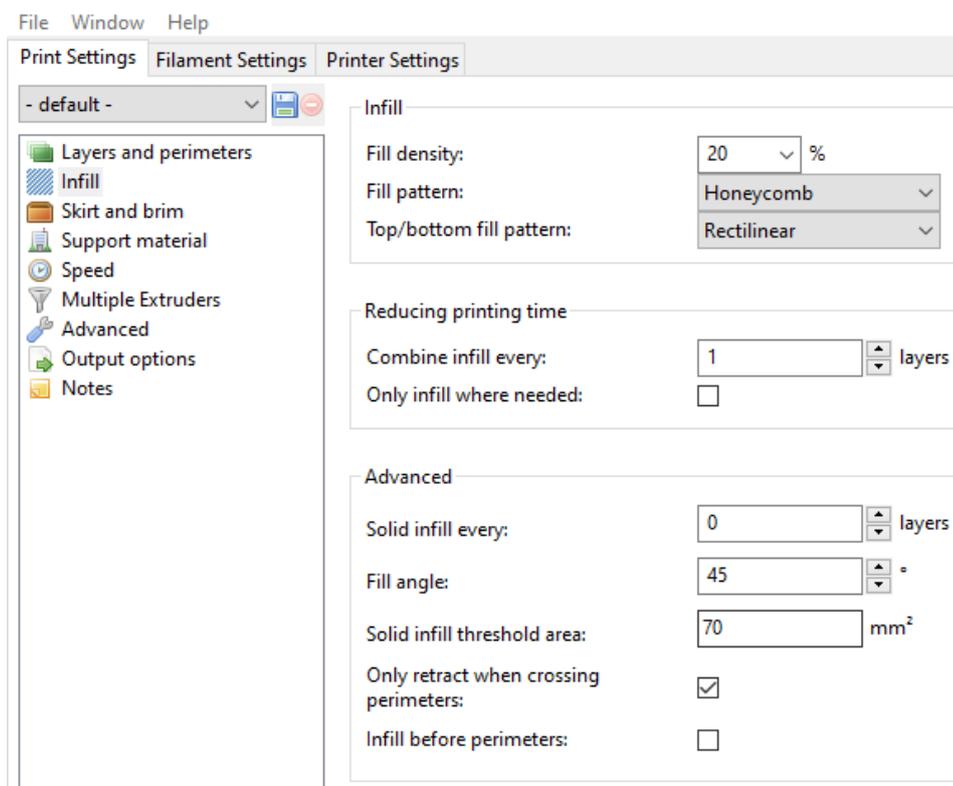


Fonte: 3DLAB, 2021.

O fatiador *Slic3r* define diferentes tipos de preenchimentos (cf. Figura 64). O *menu* de preenchimento é um dos mais importantes para o desenvolvimento de impressões tridimensionais em materiais plásticos e é necessário ser considerado para as impressões com diferentes argamassas.

Contudo, não é necessário apresentar uma grande variedade de modelos e padrões de preenchimento. O estudo apresenta apenas dois padrões.

⁷ Preenchimentos são padrões em linhas que podem variar em estilo e quantidade de material. Os tipos mais comuns são os em formato de colmeia ou linhas retas.

Figura 64 – Slic3r: Comandos de preenchimento

Fonte: o autor.

O algoritmo para impressão de argamassa define dois tipos de padrão, o padrão em zigue-zague e em grade (*Grid*). A importância desses padrões é permitir que as peças apresentem resistência mecânica e manter as camadas subsequentes sempre apoiadas. As argamassas são materiais que não apresentam a capacidade de vencer vãos durante a impressão como acontece com o plástico, *vide* Figura 65:

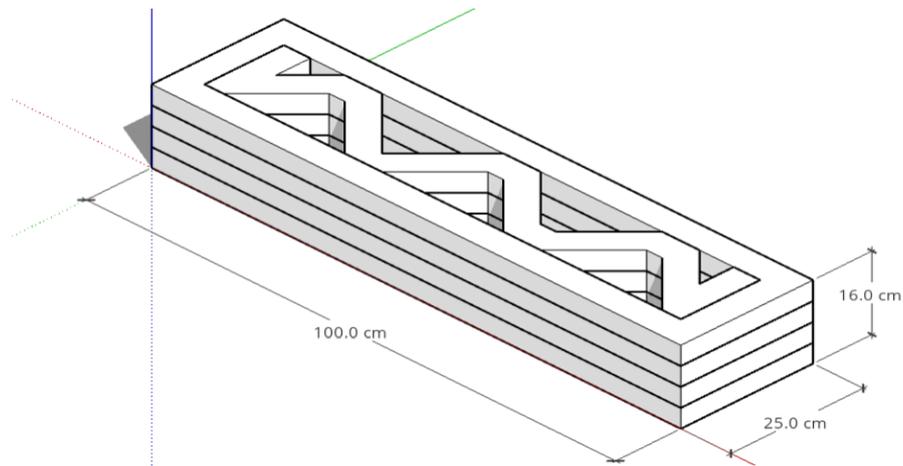
Figura 65 – Pontes impressas em impressoras tipo *desktop*

Fonte: 3DMENSIONALS, 2021.

O padrão zigue-zague é mais simples, econômico e necessita de menos tempo de impressão. O objetivo desse tipo de padrão é manter a estabilidade dos artefatos

arquitetônicos impressos. O zigue-zague é composto por linhas sobrepostas em padrões com angulação de 90° e ao final de cada linha o eixo Z sobe a espessura definida para a camada e retoma o movimento pelos perímetros internos e depois os externos, segundo pode ser visto na Figura 66:

Figura 66 – Padrão de perímetro e preenchimento em zigue-zague



Fonte: o autor.

1 algoritmo ZIGUE_ZAGUE

// O padrão zigue_zague utiliza como referência os códigos G pré-definidos para indicar a movimentação dos eixos. O algoritmo utiliza sistema de vetor de código para calcular a quantidade de pontos cartesianos que o preenchimento utiliza.

2 vetor: V [1...N] de inteiros // definição de quantos pontos são necessários para deslocar os eixos para produzir o preenchimento em zigue-zague.

3 variáveis compostas X,Y, Z: inteiro // "X , Y e Z" são variáveis referentes às coordenadas cartesianas. Z é utilizado no algoritmo como quantidade de movimentações para definir a número de camadas.

4 Constante U: real // número de coordenadas cartesianas.

5 camada: inteiro

6 **inclua** biblioteca código G // biblioteca referente aos comandos predefinidos para o código G.

7 **G01** // interpolação Linear.

8 **G17** // plano de interpolação XY.

9 **G41** // correção automática da ferramenta.

10 **G90** // sistema de coordenadas absolutas. Ativado automaticamente quando a máquina é ligada.

11 **G92** // definição da posição atual como origem. Necessário para o início do preenchimento ao final da definição do perímetro.

12 **inicio**

13 X, Y ← 0; // inicialização de variáveis.

14 Z ← 0; // inicialização constante.

15 **leia** Z ← primeira_camada // levanta o bico de extrusão de acordo com a definição da primeira camada calculada pelo algoritmo.

16 **para U de 1 ate N passo 1 faça** // estrutura de controle para definir o movimento dos eixos por meio das coordenadas cartesianas. O código lê a primeira coordenada e continua somando a próxima coordenada (1 até N).

17 **leia** X.Y [U];

18 **para** camada de 1 **ate N passo 1 faça**

19 **se** Z > 0 **faça**

20 **leia** Z (camada);

21 **senao**

22 **leia** Z← primeira_camada

23 **fimse**

24 **fimpara**

25 **fimpara**

26 **fim**

Exemplo:

G90

G92 X 0. Y.0. Z.0

Z← primeira_camada

G92 G17 G41 G01 X(1). Y(1). Z(1) // movimenta o eixo para a posição "U" de 1 até N – Primeira posição.

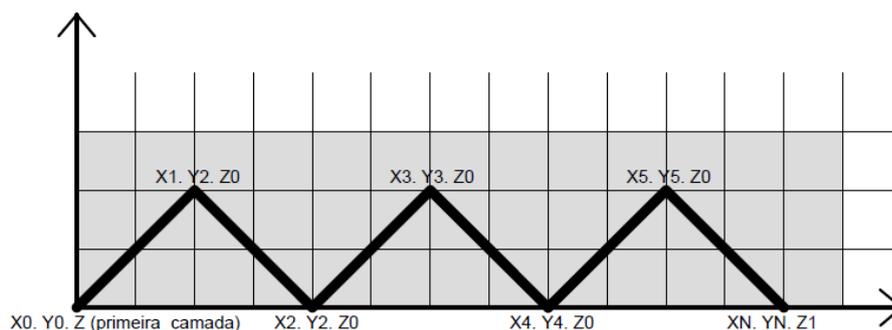
CAMADA (1) // Z equivalente 1 até N.

G01 X(1). Y(1). Z(1) // PRIMEIRA POSIÇÃO.

CAMADA (N) // indicação da última camada.

G01 X(N). Y(N). Z(N) // última posição.

Figura 67 – Gráfico de exemplo para preenchimento em zigue-zague

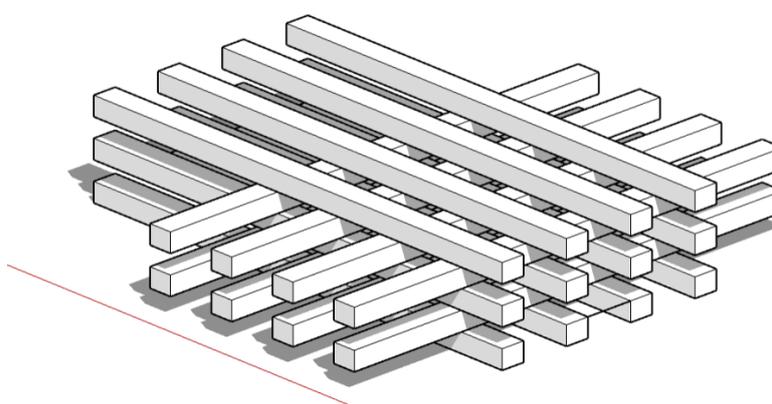


Fonte: o autor.

A grade é um padrão muito utilizado em impressões tridimensionais de plástico. Contudo, é necessário adaptar o desenho da grade para permitir um apoio adequado das camadas de preenchimento. As camadas em impressoras de mesa são feitas com espessuras que não influenciam na aderência do material.

A Figura 68 demonstra o motivo de o padrão de grade convencional não funcionar para impressões em escala 1:1. A textura de pasta do material de impressão e a altura em centímetros das camadas cria a deformação devido aos vãos entre uma camada e outra:

Figura 68 – Padrão de preenchimento em grade – espaçamento entre camada



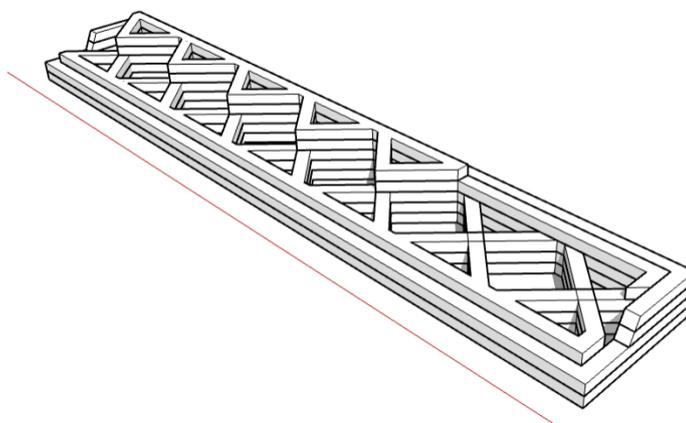
Fonte: o autor.

Nas impressoras de argamassas, para impressões em tamanho natural, as espessuras das camadas serão relevantes para a definição de impressão. Portanto, o padrão da grade deve ser feito de tal forma que permita que o preenchimento seja apoiado.

O padrão de preenchimento com grade pode ser feito com inclinação de 45° (cf. Figura 69) ou 90°. As linhas são sobrepostas para melhorar a estabilidade e garantir a aderência em todas as camadas.

Os vértices dos padrões se encontram no meio; assim, o extrusor passa pelo mesmo ponto duas vezes, permitindo uma melhor aderência. O material deve extrapolar para os lados por ter um acesso de argamassa no centro. Contudo, essa situação pode contribuir para a qualidade estrutural das edificações e não fica aparente.

Figura 69 – Padrão de preenchimento em grade com inclinação de 45°



Fonte: o autor.

1 algoritmo GRADE

// O padrão grade é similar ao zigue-zague, com a diferença de que o código define a repetição dos pontos centrais da grade. A repetição das coordenadas objetiva simplificar o código e garantir uma solidarização das camadas de preenchimento, melhorando a estruturação e estabilidade dos objetos.

2 vetor: V [1...N] de inteiros // definição de quantos pontos são necessários para deslocar os eixos para produzir o preenchimento em GRADE (cf. Figura 70).

3 variáveis compostas X,Y, Z: inteiro // “X , Y e Z” são variáveis referentes às coordenadas cartesianas. Z é utilizado no algoritmo como quantidade de movimentações para definir o número de camadas.

4 Constante L, M: real // L = número de coordenadas cartesianas nas laterais. M = número e coordenadas cartesianas no centro.

5 **inclua** biblioteca código G

6 **inicio**

7 X, Y ← 0; // Inicialização de variáveis.

8 Z ← 0; // Inicialização constante.

9 **leia** Z ← primeira_camada; // levanta o bico de extrusão de acordo com a definição da primeira camada calculada pelo algoritmo.

10 **para** L de 1 **ate** N **passo** 1 **faça**

11 **leia** X,Y [L];

12 **para** M de 1 até N **passo** 1 **faça**

13 **leia** 2 (X,Y [M]);

14 **para** camada de 1 **ate** N **passo** 1 **faça**

15 **se** Z > 0 **faça**

16 **leia** Z [camada];

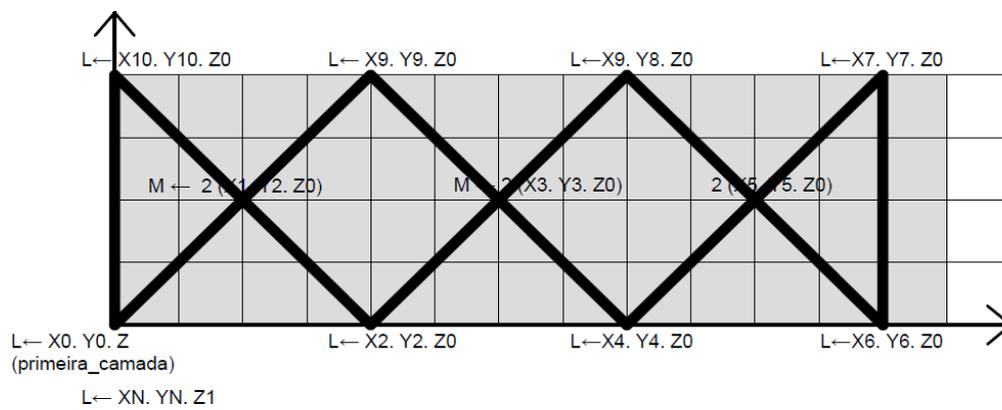
17 **senao**

```

18         leia Z← primeira_camada;
19         fimse
20     fimpara
21     fimpara
22     fimpara
23 fim

```

Figura 70 – Gráfico de exemplo para preenchimento em grade



Fonte: o autor.

7.5.6 Comandos Speed (velocidade)

Nas impressões feitas com plástico, é possível configurar a velocidade de diversos parâmetros vinculados com o movimento dos eixos e com cada parte de uma peça impressa. Contudo, a impressão de argamassa é definida no estudo com poucos parâmetros de velocidade.

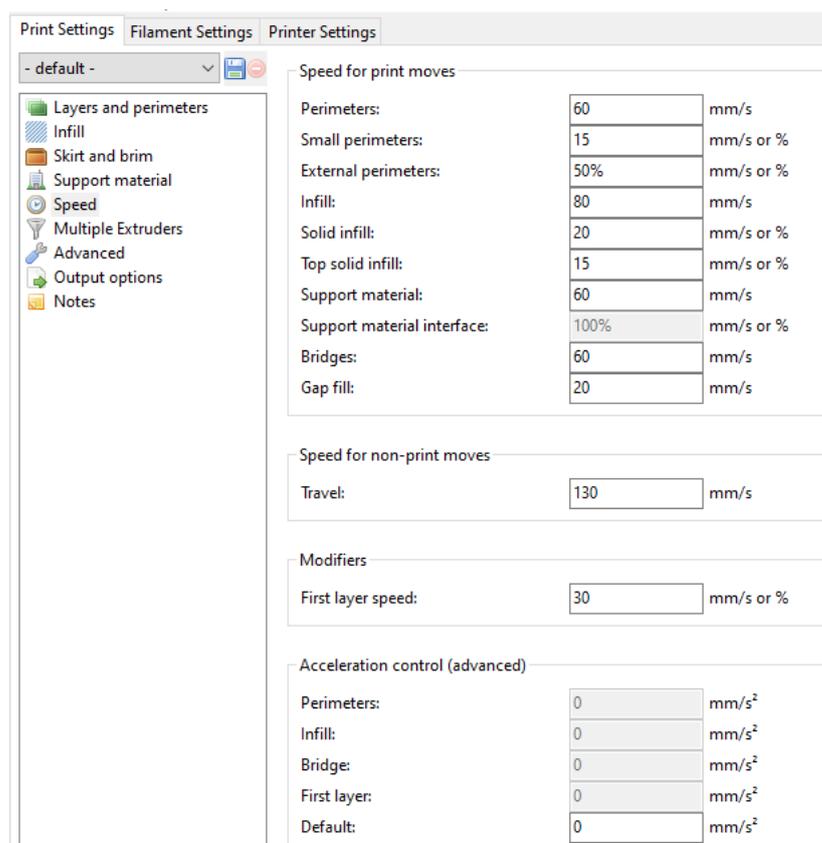
A velocidade de movimento dos eixos (cf. Figura 71) deve ser a mesma para todas as camadas, perímetros internos, externos e preenchimento. Essa simplificação visa melhorar a qualidade e a estabilidade do movimento da impressora. As velocidades dos parâmetros “perímetros”, “perímetros externos”, “preenchimento”, “camada superior sólida” (totalmente preenchida) e “primeira camada” (totalmente preenchida) devem ser definidas para centímetros por minutos (cm/min).

Os comandos *Reducing printig time* (redução do tempo de impressão) e *Advanced* (avançado) e demais parâmetros do comando “velocidade” devem ser comentados/desabilitados para simplificar o programa.

Os outros comandos relacionados ao *Print settings* (configurações de impressão) devem ser comentados/desabilitados. São comandos referentes à impressão de plástico e não se aplicam a argamassa, conforme será visto na Figura 72.

- ***Skirt and Brim*** (saia e aba): são parâmetros destinados a criar uma melhor aderência das peças de plástico à mesa de impressão por meio de uma aba de sacrifício. A primeira camada é mais larga;
- ***Support Material*** (material de suporte): nas impressões feitas com plástico, é necessário criar peças de sacrifício para criar modelos e impressões com peças em balanço ou com vãos entre apoios. Sem as peças de sacrifício, o plástico cai e deforma a impressão;
- ***Multiple Extruders*** (múltiplos extrusores): alguns modelos de impressoras tridimensionais utilizam múltiplos extrusores para imprimir peças com mais de uma cor ou com mais de um tipo de plástico;
- ***Extrusion width*** (largura de extrusão): permite ajustes finos nos parâmetros relacionados aos perímetros e preenchimentos das peças;
- ***Output options*** (opções de saída): permite mudar os nomes dos códigos G criados pelo programa; e
- ***Notes*** (anotações dos usuários): permite anotações.

Figura 71 – *Slic3r* – definição de parâmetros para controle de velocidade dos eixos

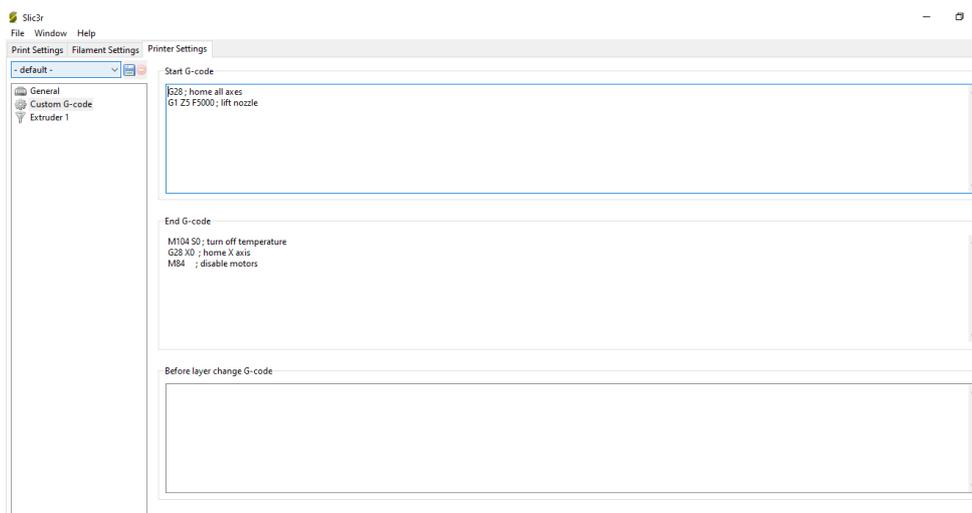


Fonte: o autor.

7.5.7 Comandos Custom G-code (Código G personalizado)

Os comandos de código G personalizado são comandos que os usuários podem escrever para que a impressora execute ações não relacionadas com o código G gerado pelo *software* para cada modelo virtual. Eles definem os movimentos da máquina antes do início da impressão e ao final da impressão, conforme demonstrado pela Figura 72:

Figura 72 – Comandos para definição de código G personalizado



Fonte: o autor.

A ferramenta *Slic3r* permite que linhas de código G sejam inseridas para definir o movimento da impressora durante o trabalho em movimentações entre as camadas.

Os blocos destinados a inserção dos comandos são:

- ***Before layer change G-code*** (antes de mudar o código G da camada);
- ***After layer change G-code*** (após a mudança de camada no código G);
- ***Between objects G-code*** (entre objetos no código G).

O *Slic3r* apresenta algumas linhas pré-configuradas que podem ajudar os usuários a entender os movimentos iniciais das máquinas e podem ser preservados para o estudo apresentado.

7.5.8 Start G-code definido para impressora FAU/UnB

G 28 Z 0;

G 28 X 0 Y 0;

G1 Z 445 F 5000;

G 92 E 0;

G1 F 1000 E5;

- **G28 Z 0** // o eixo Z se movimenta até encontrar a chave de fim de curso do eixo Z posicionado na parte de cima do pórtico. Esses movimentos foram

definidos dessa forma para evitar colisões com possíveis peças que precisem de tempo de cura e, conseqüentemente, limpeza do sistema de extrusão;

- **G28 X 0 Y0** // início dos eixos. Movimenta os eixos para as suas origens. Quando o usuário inicia a impressão, a máquina da FAU/UnB movimenta o eixo X para a direita até bater na chave de fim de curso (*endstop*). O eixo Y é representado pelo movimento do pórtilco até encontrar a chave de fim de curso.

A movimentação dos eixos com o objetivo de encontrar a chave de fim de curso e “zerar” o espaço, definir a coordenada X 0, Y 0, Z 0, garante que a máquina vai se movimentar no volume de impressão definido em projeto e no programa. A ordem de movimentação dos eixos permite que o usuário evite acidentes e possa retomar o trabalho com segurança, pois o eixo Y, movimento do pórtilco, é o último a ser deslocado para que pessoas próximas tenham tempo de reação, caso ocorram eventuais problemas.

- **G1 Z 445 F5000** // comando para descer o eixo Z e conseqüentemente o bico do extrusor, aproximando o eixo da base de impressão. G1: Movimentos lineares do eixo; Z 415: move o eixo Z descendo 445 milímetros; F 5000 – É velocidade de avanço: mm/min;
- **G 92 E 0** // o código **G 92** estabelece uma nova origem para o início da movimentação. O **E 0** é referente ao extrusor. A máquina não extruda argamassa nesse momento;
- **G1 F 1000 E5** // deposita 5 cm de argamassa para evitar movimento de impressão sem material. O comando G1 define um avanço de trabalho programado. Os eixos e o extrusor se movimentam entre dois pontos que são calculados entre as coordenadas com referência ao zero programado e com a velocidade (F) pré-definida pelo usuário.

7.5.9 End G-code definido para impressora FAU/UnB

G28 Z0

G28 X 0 Y0

G28 X 0 Y0

G1 F 1000 E500

M84

- **G28 Z0** // Ao final da impressão, o eixo Z é o primeiro a se movimentar. O eixo sobe até encontrar a chave de fim de curso para evitar colisões com os artefatos impressos;
- **G28 X 0 Y0** // Movimenta primeiro o X e depois o Y. Ambos se movimentam em direção a chave de fim de curso;

// Os eixos voltam para a origem do espaço, X 0, Y 0, Z 0.

- **G1 F 1000 E 500** // o comando visa limpar o extrusor automaticamente ao final da impressão. É necessário evitar que a argamassa fique no sistema de extrusão, ou seja, ao final da movimentação dos eixos, para zerar o sistema, o extrusor descarta o material não utilizado;
- **M84** // desliga os motores da máquina.

// O código M é referente a funções auxiliares de movimentação das máquinas tipo CNC.

7.5.10 Between objects for sequential printing (entre impressão sequencial) em formato código G (Gcode)

Pausa para cura (cf. Figura 73).

G 60 Z H

// H ← camada_final * número de camadas;

G 28 Z 0

G 28 X 0 Y 0

G1 F 1000 E 500

- **G 60 Z H** // o comando G 60 salva a posição dos eixos no momento da pausa. Z H deve ser modificado pelo usuário dependendo da necessidade. A

camada deve ser substituído pela altura no eixo Z para definir uma determinada quantidade de camadas para o processo de cura.

Exemplo:

$H \leftarrow \text{camada_final} * \text{número de camadas};$

$H \leftarrow 4,935 * 30 = 148,05 \text{ cm}$

G 60 Z 148,05

Figura 73 – Código G Pausa para cura – inserção na ferramenta *Slic3r*

```
Between objects G-code (for sequential printing)
G 60 Z 148,05 // ponto de salvamento para retomada.
// H← camada_final * número de camadas;
G 28 Z 0;
G 28 X 0 Y 0;
G1 F 1000 E 500;
```

Fonte: o autor.

8 Resultados

O estudo apresenta como resultados obtidos algoritmos originais com parâmetros e definições de funcionamento para ferramentas de impressão 3D com argamassas. Os códigos foram escritos para indicar como ferramentas digitais para impressoras 3D tipo *desktop* podem ser modificadas.

As indicações de medidas em milímetros dos programas originais não permitem uma visualização adequada de maquetes virtuais em metros. Portanto, foram apresentadas propostas de mudanças das medidas para metros e centímetros, unidades usuais para os profissionais de arquitetura e construção civil.

O desenvolvimento dos algoritmos originais relacionados diretamente às características das argamassas e da impressão em tamanho natural foram os mais interessantes e promissores para pesquisa. A quantidade de algoritmos e códigos que acreditava-se necessária no início do estudo mostrou-se menor e mais eficientes, com poucas linhas de código. Os parâmetros das argamassas vinculados à retração e à plasticidade foram considerados os mais relevantes para o trabalho e foram necessários apenas dois códigos com cálculos matemáticos relativamente simples para incorporar as características ao processo de impressão 3D.

Os algoritmos relacionados ao processo de pausa para cura foram considerados importantes para o processo de impressão em tamanho natural com argamassas. É possível fazer uma edificação ou elemento arquitetônico sem pausas ou interrupções. Contudo, o processo ininterrupto pode limitar possíveis pesquisas ou estudos de novos materiais, traços e edificações em mais de uma etapa. As pausas elaboradas para impressões em tamanho natural podem contribuir para a popularização da tecnologia e para o desenvolvimento de novas pesquisas.

Os códigos restantes foram criados com o intuito de facilitar o uso das impressoras em tamanho natural por parte dos profissionais da construção civil e limitar a quantidade de parâmetros relacionados à velocidade para evitar possíveis danos ou acidentes. Os algoritmos desenvolvidos nesta pesquisa são autorais e apresentam as definições e rotinas criadas para impressoras de tamanho natural.

A produção de testes não foi incluída no escopo do trabalho. O processo apresenta a análise das ferramentas livres e quais parâmetros devem ser modificados

e incluídos para a impressão em tamanho natural com materiais pastosos. Os testes serão executados em estudo futuro. Os códigos foram compilados nas páginas seguintes.

1 algoritmo BASE_METROS

// Apresenta a modificação das características das bases de impressão. O algoritmo visa a indicação da base em metros e é subdividida em centímetros. O código apresenta linhas comentadas para que possa ser utilizado na impressora de estudo da FAU/UnB ou para uma impressora em tamanho natural.

2 variaveis unidade X, unidade Y: inteiros; // declaração de variáveis.

3 grade ← (1 cm; 10cm)); // definição de grade dividida em centímetros 4
para impressora FAU/UnB de medidas reduzidas.

4 grade ← (10 cm ; 100cm)); // definição de grade dividida em
centímetros/metros para utilização em impressoras de tamanho natural.

5 dimensões_base: inteiros;

6 **inicio** // identificação de início do bloco correspondente ao algoritmo.

7 unidade X ← 63;

8 unidade Y ← 89;

9 // medida da impressora de base de impressão para FAU/UnB.

10 // unidade X ← 600.

11 // unidade Y ← 900; // medidas da impressora para tamanho natural.

12 dimensoes_base (unidade X, unidade Y, 0, 0, 0);

13 **fim** // fim do algoritmo proposto.

1 algoritmo MOVIMENTO_Z

// O algoritmo foi desenvolvido para facilitar a movimentação do eixo Z. O código apresenta a possibilidade de inserção de velocidade e valores pré-definidos de espaços percorridos. A velocidade e as distâncias são definidas em cm/min. O eixo deve ser em centímetros para permitir a colocação precisa do eixo Z na altura de camada desejada.

2 **variaveis** velocidade, valores_movimento: real;

3 multiplicador_distancia: inteiro;

4 **constante** home: logico;

5 **inicio**

6 **escreva** ('Insira a Velocidade Desejada');

7 **leia** (velocidade);

8 **escreva** ('velocidade= cm/min', velocidade);

9 valores_movimento (0.5, 5, 50) ;// unidades de medida em cm.

10 multiplicador_distancia (0.5 * 10); // multiplica a unidade básica de medida 10 para aumentar a distância percorrida.

11 **inicio**

12 **leia** (home); // referente ao botão com desenho de casinha.

13 **se falso** move para cima;

14 **ate verdadeiro** bater na chave de fim de curso; // interrompe o movimento na origem do plano cartesiano.

15 aguardar entrada de dados;

16 **fim**

17 **fim**

1 algoritmo MOVIMENTO_XY

// O algoritmo de movimento nos eixos X e Y foi definido para permitir uma movimentação adequada utilizando unidades em metros. O código é similar ao MOVIMENTO_Z.

2 variaveis velocidade, valores_movimento: real

3 multiplicador_distancia: inteiro

4 constante *home*: logico

5 **inicio**

6 **escreva** ('Insira a Velocidade Desejada');

7 **leia** (velocidade);

8 **escreva** ('velocidade= m/min', velocidade); // velocidade em metros por minutos.

9 valores_movimento (0.01, 0.1, 1, 5) // unidades de medida em metros.

10 multiplicador_distancia (0.01 * 10) // multiplica a unidade básica de medida para aumentar a distância percorrida.

11 **inicio**

12 **leia** (*home*); // referente ao botão com desenho de casinha.

13 **se falso** move para o lado

14 **ate verdadeiro** bater na chave de fim de curso

15 <aguarde entrada de dados>

16 **fim**

17 **fim**

1 algoritmo BICO_EXTRUSAO

// O código de bico de extrusão tem por objetivo permitir a configuração do diâmetro do bico utilizado em diferentes impressoras de argamassa. O algoritmo calcula a espessura das camadas para manter a proporção referente ao diâmetro e apresenta as possíveis espessuras de camadas mínimas e máximas.

2 variáveis diametro_bico: inteiro;

3 camada_min, camada_max: real;

4 **inicio**

5 (diametro_bico) > [0]; // diâmetro do bico em cm. O bico aparece como zero para permitir que seja modificado pelo usuário.

6 **escreva** ('Insira o Diâmetro do Bico de extrusão');

7 **leia** (diametro_bico);

8 **escreva** ('Diâmetro do Bico= cm', diametro_bico);

9 **inicio**

10 **conecta** ao bloco de fatiamento;

11 **define** ('diametro_bico', [5]); // por padrão, a ferramenta deve mostrar o bico de extrusão com 5 centímetros de diâmetro.

12 **se** diametro_bico ← 5; // ← significa igual (=).

13 **entao** ('camada_min', [1]); // camada mínima 1 centímetro.

14 ('camada_max', [5]); // camada mínima 5 centímetros.

15 **inicio**

16 **se** diametro_bico > 5;

17 **entao** ('camada_min', [(1*diametro_bico)/5]); // camada mínima proporcional ao bico de extrusão para manter a proporção centímetro.

18 ('camada_max', [(5*diametro_bico)/5]); // camada mínima proporcional ao bico de extrusão para manter a proporção centímetro.

19 **fim**

20 **fim**

21 **fim**

1 **algoritmo PAUSA_RETOMADA**

// É um dos códigos mais importantes do estudo. Foi escrito para permitir que os usuários possam interromper o movimento dos eixos e retomar a impressão sem mover o bico para pausas rápidas. Apresenta a possibilidade de pausas longas com movimentação dos eixos e definição de posição dos eixos para evitar possíveis colisões com os limites das impressoras.

// O código apresenta a pausa para cura, isto é, processo de interrupção da impressão para que o material utilizado possa ganhar resistência. A pausa para cura salva a coordenada do código G para que a impressão possa ser retomada do exato ponto de parada.

2 variaveis pausa_mov , pausa_cura, salvar_prog, pausa: logico

3 // pausa_mov para a impressão e movimenta o eixo X para direita. pausa_cura movimenta os eixos para a origem dos eixos (*home*). O comando identifica a linha de código e salva o código G sem as linhas que já foram executadas.

4 coordenada: vetor [4] // número 4 é referente às posições dos elementos que compõem um vetor, variável com vários elementos. No caso, o vetor coordenada é referente à linha do código G que indica posição "G" referente ao código indicado para o movimento da máquina; X, Y e Z são referentes ao posicionamento do extrusor no espaço. Ex.: G1 X 100 Y 750 Z 300. Não é necessário comando para extrusão (E) e para velocidade (F).

5 constante diretorio: vetor [3] // encontrar pasta.

6 **inicio**

7 **escreva** ('PAUSA COM MOVIMENTO' =,pausa_mov);

8 **se** pausa_imp falso **entao**

9 **executa** < codigo G >

10 **se** pausa_imp verdadeiro **entao**

11 **pausa** motores

```
12          // x← + 100 // move eixo X um metro para direita. Obs.: base impressora
natural 600 x 900.

13          // se 100 > X > 501 entao.

14          Y← + 100 // o algoritmo prioriza o movimento do eixo X por ser
mais leve. O eixo Y movimenta todo o pórtico da impressora. Se o eixo X estiver muito
perto do limite da impressora, deve mover o eixo Y.

15          // se Y > 801 entao.

16          Y← - 100 // eixo Y move na posição oposta para não colidir com
o fim da base de impressão.

17 // As linhas 8, 9, 10 estão comentadas (//) pois se referem a impressora de
tamanho natural.

19 se 20 > X > 43 entao
20     X← + 20
21     senao
22     X← - 20
23 se 20 > Y > 69 entao
24     Y← + 20
25     senao
26     Y← - 20
27 // As linhas 12, 13 e 14 se referem à impressora da FAU/UnB.
27 fim
28 inicio
29 escreva ('Pausa para Cura' =,pausa_cura);
30 se pausa_cura falso entao
31     {execute < codigo G >}
32 se pausa_cura verdadeiro entao
```

```
33      pausa motores
34      leia coordenada < codigo G > // salva a linha de código G para que o
sistema retome a impressão depois da cura do material.
35      X = 0, Y = 0, Z = 0; // movimenta os eixos para a origem do plano
cartesiano.
36      escreva ('Retomo deste ponto = [coordenada]', coordenada);
37      escreva ('Salvar Progresso= ', salvar_prog);
38      se salvar_prog falso entao
39          leia coordenada;
40      se salvar_prog verdadeiro entao
41          leia coordenada em diretorio = (C:) desktop < Gcode: CURA; > //
buscar a linha de código G em um diretório na área de trabalho do computador.
42      leia < diretorio >;
43      fim
```

```
1  algoritmo VELOCIDADE_SEM_EXTRUSAO
2  // O bloco de velocidade sem extrusão permite que os eixos se movimentem
entre os pontos de impressão. Esses pontos podem variar para definir vãos ou
diferenças de alturas dos elementos arquitetônicos.
3
4  // A velocidade é definida utilizando como parâmetro a plasticidade da
argamassa. O usuário insere os limites de liquidez e plasticidade e o algoritmo
apresenta velocidades adequadas para a movimentação. Esse algoritmo é definido
com a plasticidade para evitar que a argamassa escorra pelo bico durante o processo
de movimentação dos eixos.
5
6  variaveis coordenada1, coordenada2, var_tempo, vel_semE: real; // definida
em metros por minuto (m/min).
7
8  plasticidade, LL, LP: real; // LL = Limite de Liquidez e LP = Limite de
Plasticidade
9
10 inicio função vel_semE;
11 leia (coordenada1);
12 leia (coordenada2);
13
14 vel_semE ← (coordenada1 - coordenada2)/ var_tempo; // definição de
velocidade de movimento
15
16 retorne vel_semE;
17
18 fimfunção
19
20 inicio
21 escreva ('Qual é o Limite de Liquidez= ', LL);
22 leia (LL); // O usuário insere o Limite de Liquidez.
23
24 escreva ('Qual é o Limite de Plasticidade= ', LP);
```

```

17  leia (LP); // o usuário insere o Limite de Plasticidade.
18  plasticidade ← (LL – LP); // o índice de plasticidade é definido pela diferença
entre o Limite de Liquidez (LL) e o Limite de Plasticidade (LP).
19  // É sugerido escolha de argamassas com índice de plasticidade entre 4% a 11
% (MATUK e HAY apud NEVES, FARIA, 2011).
20      inicio
21  se ( 7 >= plasticidade >= 11 );
22      entao vel_semE = 3 ou vel_semE = 4 vel_semE = 5 // se a plasticidade
da argamassa for maior ou igual a 7% e menor ou igual a 11%, permite a inserção de
três velocidades predefinidas.
23  // FAU/UnB e dimensões reduzidas usar parâmetros reduzidos em 100%.
24  // Linhas comentadas para serem substituídas para o uso da impressora de
menor dimensão da FAU/UnB.
25      // entao vel_semE = 0.3. ou vel_semE = 0.4 vel_semE = 0.5.
26      escreva ('Velocidades permitidas 3 ou 4 ou 5 m/min = ' vel_semE); //
permite que o usuário insira a velocidade desejada.
27      // escreva ('Velocidades permitidas 0.3 ou 0.4 ou 0.5 m/min = '
vel_semE).
28      enquanto vel_semE <> 3 ou 4 ou 5 faça
29      // enquanto vel_semE <> 0.3 ou 0.4 ou 0.5 faça.
30      leia ( vel_semE);
31      senao
32      se ( 4 >= plasticidade >= 6.9 );
33      entao vel_semE = 1 ou vel_semE = 2 vel_semE = 2.90 // se a
plasticidade da argamassa for maior ou igual a 4% e menor ou igual a 6,9%, permite
a inserção de três velocidades predefinidas.
34  // impressora FAU/UnB com dimensões reduzidas. Os parâmetros são
reduzidos em 100%

```

```
35 // linhas comentadas para serem substituídas para o uso da impressora de
    menor dimensão da FAU/UnB.

36 // entao vel_semE = 0.1 ou vel_semE = 0.2 vel_semE = 0.29

37 escreva ('Velocidades permitidas 1 ou 2 ou 2.90 m/min = ' vel_semE);
// permite que o usuário insira a velocidade desejada.

38 // escreva ('Velocidades permitidas 0.1 ou 0.2 ou 0.29 m/min =
'vel_semE).

39 enquanto vel_semE <> 1 ou 2 ou 3 faça

40 // enquanto vel_semE <> 0.1 ou 0.2 ou 0.3 faça.

41 leia ( vel_semE)

42 fim

43 fimse

44 fimse

44 fim
```

1 **algoritmo VOLUME_METROS**

// O código apresenta a definição de valores relativos aos eixos X, Y e Z em metros e subdivididos em centímetros. O algoritmo foi escrito para permitir que o usuário modifique os espaços de movimentação dos eixos. Contudo, foram definidos limites e o algoritmo avisa ao usuário se os dados inseridos são adequados ou não e só continua o funcionamento se os dados forem inseridos de forma correta.

2 variaveis unidade X, unidade Y, unidade Z: inteiros;

3 // grade←(1 cm; 10 cm); // definição de grade dividida em centímetros para impressora FAU/UnB de medidas reduzidas.

4 // subgrade←(10 mm por 10 mm); // a base deve apresentar subdivisões em milímetros para impressora FAU/UnB de medidas reduzidas.

5 grade←(10 cm; 100 cm); // definição de grade dividida em centímetros/metros para utilização em impressoras de tamanho natural.

6 subgrade←(100 cm por 100 cm); // A base deve apresentar subdivisões em centímetros para facilitar a organização da impressão.

7 dimensao_volume: inteiros

8 **inicio**

9 **escreva** ('Qual é a medida da base em X= ',unidade X);

10 **leia** (unidade X);

11 **se enquanto** unidade X > 600 **faça**

12 // **se enquanto** unidade X > 63 **faça** // medida de X para a impressora reduzida da FAU/UnB. Linha comentada para não influenciar o funcionamento do algoritmo.

13 **escreva** ('Medida inválida. A impressora não pode ter mais de 6 metros de largura');

14 // **escreva** ('Medida inválida. A impressora não pode ter mais de 63 centímetros de largura'); // impressora reduzida da FAU/UnB.

```
15  escreva ('Qual é a medida da base em Y ' = unidade Y);
16  leia (unidade Y);
17  se enquanto unidade Y > 900 faça
18  // se enquanto unidade Y > 89 faça // impressora reduzida da FAU/UnB.
19  escreva ('Medida inválida. A impressora não pode ter mais de 9 metros de
comprimento');
20  // escreva ('Medida inválida. A impressora não pode ter mais de 89 centímetros
de comprimento'); // impressora reduzida da FAU/UnB.
21  escreva ('Qual é a medida da base em Z ' = unidade Z);
22  leia (unidade Z);
23  se enquanto unidade Z > 400 faça
24  // se enquanto unidade Z > 41,5 faça // impressora reduzida da FAU/UnB.
25  // escreva ('Medida inválida. A impressora não pode ter mais de 41,5
centímetros de Altura');
26  escreva ('Medida inválida. A impressora não pode ter mais de 4 metros de
Altura');
26  dimensao_volume← (unidade X, unidade Y, unidade Z);
27  leia (unidade X, unidade Y, unidade Z, 0, 0, 0);
28  escreva ('Base = "unidade X", 'por', "unidade Y", unidade X, Unidade Y');
29  escreva ('Altura Máxima para impressão = ', unidade Z);
30          fim
31          fimse
32          fimse
33          fimse
34  fim
```

1 **algoritmo RETRAÇÃO**

// O parâmetro de retração foi considerado importante para o estudo. Algumas argamassas e traços podem apresentar essa característica e variações de dimensões.

// O código permite que o usuário insira o limite de retração e apresente por meio de mensagens se a argamassa pode apresentar problemas como rachaduras. O programa utiliza os dados relacionados ao bico de extrusão e espessuras de camadas e compensa automaticamente o efeito gerado pela retração.

2 **inclua** biblioteca Matematica ← mat; // permite arredondar resultados diminuindo as casas decimais.

3 variaveis LR, retracao, camada, altura_objeto, camada_final: inteiros; // LR - significa limite de retração.

4 // A altura do objeto é referente à altura do elemento arquitetônico representado de forma virtual. Exemplos: paredes, pilares, bancos.

5 altura_retracao, quantidade_camadas: real;

6 **inicio**

7 **escreva** ('Qual é o Limite de retração= ', LR);

8 **leia** (LR); // o usuário insere o limite de retração aferido.

9 retracao ← LR /100;

10 **se** LR <= 2/100 **entao**

11 **escreva** ('Altura de camada= cm', camada); // usuário insere a altura de camada desejada.

12 **se** LR > 2/100 **entao**

13 **escreva** ('É possível que sua impressão apresente rachaduras');

14 **leia** (camada);

15 **inicio**

29 **fimse**

30 **fin**

31 **fin**

```
1  algoritmo PLASTICIDADE_VELOCIDADE
2  // Neste algoritmo são apresentadas linhas utilizadas em outros algoritmos pois
   são necessárias para definir parâmetros em partes diferentes do funcionamento da
   ferramenta.
   // O algoritmo é similar ao criado para velocidade sem extrusão. O usuário pode inserir
   os dados e características das argamassas e o código realiza os cálculos referentes
   à plasticidade. O programa utiliza como base as velocidades utilizadas no algoritmo
   VELOCIDADE_SEM_EXTRUSAO e, dependendo do índice, calcula uma
   compensação da velocidade com o objetivo de garantir a movimentação adequada
   dos eixos e a qualidade da impressão.
3  variaveis diametro_bico, vel_imp: inteiro; // vel_imp = velocidade de impressão
   dos modelos ou artefatos arquitetônicos.
4  plasticidade, LL, LP: reall; // LL = Limite de Liquidez e LP = Limite de
   Plasticidade.
5  inicio
6  (diametro_bico) > [0]; // diâmetro do bico em cm. O bico aparece como zero
   para permitir que seja modificado pelo usuário.
7  escreva ('Diâmetro do Bico= cm', diametro_bico);
8  leia (diametro_bico);
9  escreva ('diametro_bico', [5]); // por padrão, a ferramenta deve mostrar o bico
   de extrusão com 5 centímetros de diâmetro.
10 inicio
11 escreva ('Qual é o Limite de Liquidez= ', LL);
12 leia (LL); // o usuário insere o Limite de Liquidez.
13 escreva ('Qual é o Limite de Plasticidade= ', LP);
14 leia (LP); // o usuário insere o Limite de Plasticidade.
```

1 algoritmo PRIMEIRA_CAMADA

// A camada inicial é calculada por último, pois utiliza a definição de retração, e, conseqüentemente, a espessura das demais camadas do objeto. A primeira camada sofre uma diminuição de 30% para ajudar na aderência à base de impressão. O objetivo é garantir uma camada inicial homogênea, o que contribui para a qualidade das próximas camadas.

2 variaveis diametro_bico, camada_final, primeira_camada: inteiro;

3 **inicio**

4 **busca** BICO_EXTRUSAO; // utiliza o algoritmo que foi definido para o bico de extrusão.

5 **busca** RETRAÇÃO; // utiliza o algoritmo que foi definido para o bico de retração.

6 primeira_camada ← camada_final / 1.3; // a primeira camada é aproximadamente 30% menor que as demais camadas que são definidas pela variável camada_final.

7 // Exemplo: $306 / 62 = 4,935 / 1.3 = 3,796$ cm de espessura.

8 **fimbusca**

9 **fimbusca**

10 **fim**

1 algoritmo ZIGUE_ZAGUE

// O padrão zigue_zague utiliza como referência os códigos G pré-definidos para indicar a movimentação dos eixos. O algoritmo utiliza sistema de vetor de código para calcular a quantidade de pontos cartesianos que o preenchimento utiliza.

2 vetor: V [1...N] de inteiros; // definição de quantos pontos são necessários para deslocar os eixos para produzir o preenchimento em zigue-zague.

3 variáveis compostas X,Y, Z: inteiro; // "X , Y e Z" são variáveis referentes às coordenadas cartesianas. Z é utilizado no algoritmo como quantidade de movimentações para definir a número de camadas.

4 Constante U: real; // número de coordenadas cartesianas.

5 camada: inteiro;

6 **inclua** biblioteca código G // biblioteca referente aos comandos predefinidos para o código G.

7 **G01** // interpolação linear.

8 **G17** // plano de interpolação XY.

9 **G41** // correção automática da ferramenta.

10 **G90** // sistema de coordenadas absolutas. Ativado automaticamente quando a máquina é ligada.

11 **G92** // definição da posição atual como origem. Necessário para o início do preenchimento ao final da definição do perímetro.

12 **inicio**

13 X, Y ← 0; // inicialização de variáveis.

14 Z ← 0; // inicialização constante.

15 **leia** Z ← primeira_camada; // levanta o bico de extrusão de acordo com a definição da primeira camada calculada pelo algoritmo.

16 **para U de 1 até N passo 1 faça** // estrutura de controle para definir o movimento dos eixos por meio das coordenadas cartesianas. O código lê a primeira coordenada e continua somando a próxima coordenada (1 até N).

17 **leia** X.Y [U];

18 **para** camada de 1 até N **passo 1 faça**

19 **se** Z > 0 **faça**

20 **leia** Z (camada);

21 **senao**

22 **leia** Z← primeira_camada;

23 **fimse**

24 **fimpara**

25 **fimpara**

26 **fim**

Exemplo:

G90

G92 X 0. Y.0. Z.0

Z← primeira_camada

G92 G17 G41 G01 X(1). Y(1). Z(1) // movimenta o eixo para a posição "U" de 1 até N – Primeira posição.

CAMADA (1) // Z equivalente 1 até N.

G01 X(1). Y(1). Z(1) // PRIMEIRA POSIÇÃO.

CAMADA (N) // indicação da última camada.

G01 X(N). Y(N). Z(N) // última posição.

1 algoritmo GRADE

// O padrão grade é similar ao zigue-zague com a diferença que o código define a repetição dos pontos centrais da grade. A repetição das coordenadas objetiva simplificar o código e garantir uma solidarização das camadas de preenchimento, melhorando a estruturação e estabilidade dos objetos.

2 vetor: V [1...N] de inteiros; // definição de quantos pontos são necessários para deslocar os eixos para produzir o preenchimento em GRADE (cf. Figura 74).

3 variáveis compostas X,Y, Z: inteiro; // "X , Y e Z" são variáveis referentes às coordenadas cartesianas. Z é utilizado no algoritmo como quantidade de movimentações para definir o número de camadas.

4 Constante L, M: real; // L = número de coordenadas cartesianas nas laterais. M = número de coordenadas cartesianas no centro.

5 **inclua** biblioteca código G;

6 **inicio**

7 X, Y ← 0; // inicialização de variáveis.

8 Z ← 0; // inicialização constante.

9 **leia** Z ← primeira_camada; // levanta o bico de extrusão de acordo com a definição da primeira camada calculada pelo algoritmo.

10 **para** L de 1 **ate** N **passo** 1 **faça**

11 **leia** X,Y [L];

12 **para** M de 1 até N **passo** 1 **faça**

13 **leia** 2 (X,Y [M]);

14 **para** camada de 1 **ate** N **passo** 1 **faça**

15 **se** Z > 0 **faça**

16 **leia** Z [camada];

17 **senao**

```
18         leia Z← primeira_camada;
19         fimse
20         fimpara
21     fimpara
22 fimpara
23 fim
```

9 Conclusão

A pesquisa permitiu observar que o tema relacionado à impressão 3D de tamanho natural com materiais pastosos não é um campo de estudo muito explorado. Existem poucas impressoras de tamanho natural imprimindo com argamassa de cimento ou de terra e não foram encontradas ferramentas pagas ou de código aberto para impressão de grande formato.

Esta tese apresentou, como contribuição original, os principais parâmetros e algoritmos para indicar o funcionamento adequado das ferramentas digitais especializadas para impressão tridimensional com extrusão de argamassa.

O trabalho contribuiu, de forma original, ao apresentar os algoritmos para a criação de soluções de código aberto para arquitetura e o desenvolvimento de códigos abertos disponibilizados para outros pesquisadores. A tese auxilia, também de maneira original, na pesquisa de impressão com argamassa de terra e outros materiais pastosos.

O estudo e a construção da máquina, somados às pesquisas sobre os diferentes tipos de argamassas e técnicas de impressão tridimensional, direcionaram a especificação dos parâmetros escolhidos para o desenvolvimento dos algoritmos apresentados.

A definição de parâmetros e requisitos para o uso das ferramentas digitais foram propostas para a utilização por profissionais e estudantes de arquitetura e da construção civil. Os algoritmos desenvolvidos nesta tese e aqui apresentados visam facilitar o uso das ferramentas digitais e das impressoras 3D por pessoas com pouca experiência ou mesmo leigos no uso de máquinas de fabricação ou construção digital.

A tecnologia *open-source* permite a popularização da técnica de impressão 3D e o avanço da ciência, pois outros pesquisadores podem utilizar os algoritmos aqui formulados para fazer as modificações que forem necessárias para projetos futuros. O conhecimento distribuído de forma livre e gratuita é uma vantagem para as Universidades e Faculdades Brasileiras que podem colaborar entre si para o desenvolvimento de novas tecnologias.

A Quarta Revolução Industrial apresenta a possibilidade de incorporar técnicas de fabricação digital na construção civil. Os países que são potências econômicas e

tecnológicas, como os Estados Unidos da América, Japão, Alemanha e China, estão investindo em centros de pesquisa para se manterem como desenvolvedores de inovação tecnológica. É relevante ao Brasil e aos centros de pesquisa brasileiros que as tecnologias associadas essa revolução industrial sejam estudadas e implementadas no parque industrial para colocar o país em papel de destaque no panorama internacional.

O cenário apresentado orientou a questão norteadora e o objetivo principal para o trabalho, ou seja, como definir requisitos e como criar algoritmos para o desenvolvimento de ferramentas de código aberto para impressão 3D de edificações com extrusão de materiais pastosos. Inicialmente, foram traçados como objetivos específicos o estudo de parâmetros relacionados às argamassas e aos *softwares* de código aberto escolhidos para o trabalho.

O desenvolvimento de uma impressora projetada e construída nos laboratórios EdiQuali e LFDC da FAU/UnB, com dimensões de 89 centímetros de comprimento, 63 centímetros de largura e 41,5 centímetros de altura, em parceria com o pesquisador e professor Dr. Márcio Albuquerque Buson (FAU/UnB), permitiu fazer testes de movimentação e avaliar os parâmetros mais importantes para o estudo.

A construção da máquina permitiu o desenvolvimento de uma programação inovadora que define os movimentos de uma impressora de tamanho natural planejada para evitar acidentes e colisões com artefatos impressos.

A análise dos programas e códigos disponibilizados para modificação no *site* github.com permitiu traçar uma estratégia para o desenvolvimento dos algoritmos e requisitos. A linguagem definida para a criação dos algoritmos foi o Portugol.

O Portugol, como descrito e enfatizado, permite que qualquer desenvolvedor utilize os algoritmos para implementar as modificações estipulados para os *softwares*, não importando a linguagem de programação utilizada. Outro aspecto relevante foi a implementação de fluxogramas dos algoritmos desenvolvidos para apresentar uma solução visual e de fácil compreensão para desenvolvedores de ferramentas digitais.

Esta tese apresentou, ainda, como contribuição nova ao conhecimento, algoritmos para definição de base de impressão em metros, algoritmo BASE_METROS, pensado para profissionais de construção civil que estão mais acostumados a trabalhar com medidas em metros e centímetros.

O algoritmo MOVIMENTO_Z descreve a movimentação do eixo Z em centímetros por minuto (cm/min). O código define a possibilidade de inserção da velocidade desejada para a movimentação com medidas pré-definidas para garantir um controle adequado do posicionamento do extrusor. O algoritmo MOVIMENTO_XY tem o mesmo princípio do MOVIMENTO_Z com a diferença de que a medida de velocidade é definida em metros por minuto (m/min) e a base de impressão em metros.

O código referente ao bico extrusor, BICO_EXTRUSAO, foi um dos blocos necessários para o estudo. A investigação dos parâmetros das argamassas e a produção dos algoritmos apresentou a possibilidade de simplificar o sistema de tal forma que o algoritmo permite que o usuário defina o diâmetro do bico de extrusão e o código apresenta as espessuras de camada possíveis, camadas mínimas e máximas.

Outro bloco que se mostrou importante durante o trabalho foi o de pausa e retomada de impressão, PAUSA_RETOMADA. A impressão com argamassas em tamanho natural apresenta a necessidade de definir pausas durante o processo, tanto para agregar outros materiais, como armações de ferro, quanto para verificar se a impressão está ficando de acordo com o esperado, ou, ainda, para permitir a cura do material. Apesar de ser um bloco de algoritmo extenso, o seu desenvolvimento reduziu a quantidade de blocos de códigos. O algoritmo foi escrito para permitir pequenas interrupções e a retomada automática da impressão. O código especifica a pausa para o processo de cura interrompendo a impressão com movimentos de eixos pré-definidos no programa.

O fator mais importante para a cura do material é o processo de salvamento da coordenada cartesiana do código G no momento da pausa para que a impressão seja retomada na mesma posição. O algoritmo de pausa é associado aos códigos G definidos para o estudo. Os códigos reforçam o movimento da máquina para zerar os eixos e para expelir a argamassa restante no sistema de extrusão, evitando que o material endureça dentro da máquina.

Os algoritmos VELOCIDADE_SEM_EXTRUSAO e PLASTICIDADE_VELOCIDADE são relevantes, pois o estudo apresentou a necessidade de criar blocos de controle da velocidade vinculada à plasticidade dos materiais. Os códigos permitem que o usuário insira os valores referentes ao limite de

liquidez e ao limite de plasticidade e automaticamente calcula a plasticidade do material. O programa indica para o usuário se a argamassa está dentro dos limites indicados para uma boa construção. O algoritmo VELOCIDADE_SEM_EXTRUSAO foi desenvolvido para apresentar velocidades predefinidas, a fim de garantir uma movimentação adequada da impressora em trechos sem extrusão de argamassa. A movimentação é lenta devido ao peso da impressora em tamanho natural. O objetivo é evitar vibrações excessivas na máquina ou o deslocamento fora dos pontos cartesianos estipulados no código G.

O bloco PLASTICIDADE_VELOCIDADE é baseado no algoritmo de movimento VELOCIDADE_SEM_EXTRUSAO, mantendo os parâmetros de diâmetro do bico de extrusão e dos parâmetros vinculados à plasticidade. O código utiliza a velocidade definida pelo usuário e automaticamente a reduz em 10% ou a mantém, dependendo do tipo de argamassa utilizada. Esse código é importante para garantir uma boa adesão da argamassa em todas as camadas.

O algoritmo relacionado à retração apresenta os cálculos vinculados ao limite de retração de uma argamassa e como deve ser compensado para que o código corrija uma possível diminuição da espessura das camadas. O algoritmo reduziu a quantidade de blocos que se acreditava necessária no início do estudo.

Outro código importante foi o referente à primeira camada. É necessário para a impressão que a primeira camada tenha uma boa aderência à base de impressão, permitindo que as camadas subsequentes sejam estáveis. O código apresentado é um dos mais simples pois usa como base os algoritmos BICO_EXTRUSAO e RETRAÇÃO para calcular a primeira camada com uma espessura 30% menor em comparação à das demais camadas. A redução serve para que a impressora pressione a argamassa à base.

A análise das ferramentas digitais escolhidas para o estudo permitiu a identificação dos sistemas de preenchimentos das peças impressas. Os programas de impressão 3D apresentam uma grande variedade de padrões de preenchimento. Os algoritmos desenvolvidos limitam o preenchimento em apenas dois padrões, zigue-zague e grade. A definição de dois padrões tem por objetivo facilitar o entendimento e utilização da impressora 3D por usuários menos experientes.

O preenchimento com poucos padrões leva em consideração o material de impressão. A argamassa precisa estar sempre apoiada para garantir a qualidade e a estabilidade das peças impressas. Portanto, padrões com formas complexas ou com partes que ficam suspensas no ar foram desconsiderados para o estudo.

A impressão tridimensional com argamassa não necessita de partes que esquentem, o que simplifica os códigos e a construção das impressoras. A impressão com argamassa não precisa de parâmetros vinculados à definição de velocidades para imprimir pontes ou peças de suporte. O material abordado não tem o mesmo comportamento do plástico utilizado nas impressoras 3D de mesa.

As partes dos programas consideradas desnecessárias para a impressão com materiais pastosos podem ser desligadas no código fonte inserindo duas barras (//) para que sejam interpretadas como comentários dos programadores ou definindo os parâmetros com valores mínimos ou zero.

Diversos parâmetros das ferramentas podem ser mantidos sem alteração por permitirem a adaptação do uso para a impressão com argamassa. Um dos parâmetros identificados e que se mostrou útil para a impressão com argamassa é a ferramenta de personalização dos códigos G de inicialização, movimento entre camadas e finalização dos movimentos da impressora.

O código G define de forma predeterminada o movimento da máquina por meio de comandos alfanuméricos que são tabelados e reconhecidos pela programação por meio da biblioteca de códigos.

A produção de testes não é um objetivo da pesquisa. A análise das ferramentas livres e a definição de parâmetros para materiais pastosos foi especificado para possibilitar novo estudos e contribuir para popularização da fabricação com a manufatura aditiva.

O trabalho apresenta dois tipos de anexos; os fluxogramas dos algoritmos foram desenvolvidos para auxiliar o entendimento por parte de arquitetos e engenheiros.

Um dos anexos apresenta um exemplo de código G desenvolvido para demonstrar como é um código que gerencia impressão 3D. O código não foi apresentado completo pois um arquivo simples tem mais de 58.000 linhas, que

representam coordenadas cartesianas, velocidade de movimentação dos eixos e quantidade de material que deve ser extrudado em cada trecho da produção. O exemplo apresenta excertos da programação com indicação de função para cada código predefinido e partes que indicam mudança de altura para início de uma nova camada.

A pesquisa apresentou desafios para sua conclusão. O principal problema percebido foi a falta de comentários na programação das ferramentas selecionadas para o estudo. É importante que programadores deixem recados e comentários para si ou para outros programadores dentro do código para auxiliar em futuras modificações ou melhorias. Durante a pesquisa, foi necessário abrir todos os blocos de programas para verificar o funcionamento das ferramentas e como poderiam afetar os algoritmos propostos.

Outro desafio foi a presença de mais de uma linguagem de programação para o funcionamento das ferramentas. A lógica de programação associada ao funcionamento dos computadores e do desenvolvimento de algoritmos é a mesma para diferentes linguagens de programação. Contudo, a sintaxe de cada língua utiliza símbolos e construções diferentes, o que exigiu um trabalho de pesquisa para entender o funcionamento das linguagens apresentadas no estudo.

Comprovou-se que com o uso de ferramentas digitais de código aberto e o desenvolvimento de algoritmos para definição de novas funcionalidades de uso é possível agregar características dos materiais pastosos ao sistema de impressão. O uso das ferramentas pode ser simplificado por meio dos algoritmos, cálculos e parâmetros que facilitam o uso das impressoras 3D com materiais pastosos.

A pesquisa apresenta potencial de contribuir para a popularização da técnica, permitindo acesso a parâmetros para materiais e ferramentas de código aberto para impressão 3D.

10 Estudos futuros e recomendações

O trabalho abre a possibilidade para o início de outros projetos e estudos. Contudo, é importante ressaltar que a execução de testes não foi incluída no escopo do trabalho. Os testes para incluir os algoritmos ao processo de imprimir com argamassas deve ser o primeiro estudo feito após a conclusão do doutorado, utilizando impressoras em tamanho natural e reduzido.

Toda a sociedade e o meio acadêmico poderão se beneficiar dos resultados apresentados. As investigações a respeito do desenvolvimento de uma plataforma de código aberto para impressão 3D de argamassa têm grande potencial para ampliar o acesso à impressão de tamanho natural com materiais pastosos.

As técnicas de construção digital em escala 1:1 apresentadas no estudo indicam que é possível imprimir edificações inteiras em horas ou poucos dias, diminuindo o tempo de construção.

O trabalho focou no desenvolvimento dos algoritmos e requisitos de uso das ferramentas para impressão tridimensional. Portanto, não foi possível definir o tempo ou a quantidade de argamassa necessária para imprimir artefatos arquitetônicos.

A impressão 3D com argamassa permite investigações para aprimorar os algoritmos destinados para o funcionamento das máquinas. Pesquisas referentes a diferentes traços de argamassas e o uso de aditivos para melhorar o desempenho do material impresso, o tempo necessário para a impressão de diferentes modelos de objetos ou mesmo de edificações, testes de esforço com diferentes modelos e quantidade de preenchimento para definir a resistência de peças impressas e, principalmente, como esses parâmetros podem ser incorporados na programação devem ainda ser estudados.

A partir da conclusão deste trabalho, é apresentada a necessidade de ampliar o estudo e produzir testes e ensaios com as impressoras tridimensionais de materiais pastosos. A produção integrada da programação, associada ao funcionamento das máquinas e ao desenvolvimento de traços de argamassas especiais para impressão tridimensional é uma necessidade para integrar a pesquisa brasileira ao campo da construção digital na Quarta Revolução Industrial.

Outra conclusão importante é a possibilidade de ampliar os estudos relacionados à impressão de edificações em tamanho natural para definir custos de produção relacionados ao preço dos materiais, à mão de obra e à fabricação junto ao deslocamento das impressoras 3D para definir se a tecnologia pode ser utilizada para diminuir o problema do déficit habitacional brasileiro.

O desenvolvimento da impressora 3D nos laboratórios da Faculdade de Arquitetura e Urbanismo da Universidade de Brasília, somados com as pesquisas referentes aos parâmetros das ferramentas e argamassas, permitiu o desenvolvimento dos algoritmos inovadores apresentados na pesquisa. Contudo, os estudos são preliminares diante das possibilidades que a tecnologia apresenta para a produção arquitetônica.

REFERÊNCIAS

- ALBINO, J. S.; FONTANELA, C.; SANTOS, M. *A sociedade 5.0 como instrumento de promoção dos direitos sociais no Brasil*. *Justiça do Direito*, v. 34, n. 1, p. 29–56, 2020.
- ALEKSANDROVICH, M.; DEREVYANKIN, T.; DMITRY, D.; LEONIDOVICH, R. NIKOLAEVICH, S.; PAVLOVICH, G.; SAAK, A.; VADIMOVICH, T.; VASILEYEV, R.; VYACHESLAVOVICH, G. *The use of 3D-Printing in construction*. Moscow State University of Civil Engineering, 2019.
- ALLOUZI, R.; AL-AZHARI, W.; ALLOUZI, R. *Conventional Construction and 3D printing: A comparison Study on Material Cost in Jordan*. *Journal of Engineering*. Hindawi, 2020.
- ALMEIDA, R, S. *Aprendendo algoritmos com VisuAlg*. Rio de Janeiro: Ed. Ciência Moderna, 2013.
- ALVES, J. L.; NETO, J. B. *A Prototipagem Rápida na Indústria Nacional*. [s.l.] Resarchgate.net, 2015.
- ANGELO, A.; NEVES, H.; DE CAMPOS, P. E. F. (2012). *Fab Lab Kids: Oficina de projetos socioambientais para crianças de escolas públicas fazendo uso da eletrônica e da fabricação digital*. Trabalho apresentado no XVI Congresso da Sociedade Iberoamericana de Gráfica Digital, Fortaleza, Brasil. Disponível em: <http://cumincades.scix.net/data/works/att/sigradi2012_404.content.pdf>. Acesso em: 12 jul. 2019.
- ARDUINO, CC., 2021. Disponível em: <<https://www.arduino.cc/en/software>>. Acesso em: 23 mar. 2020.
- ASHOK, G.; VENKATESWARULU, M.; SURESH, K. *Establishment of low cost homes using SuperAdobe Technology*. 2016.
- ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. *NBR 6502: Rochas e Solos*. 1995.
- BALZANI, R. LabMaker IESB. CAUUIA20241RIO. Disponível em: <<https://caubr.gov.br/expouia2021rio/labmaker-iesb/>>. Acesso em: 13 set. 2021.
- BALZANI, R.; FURTADO, N. S. A produção de impressoras tridimensionais de baixo custo para estudantes de arquitetura. Dissertação de mestrado—Brasília: Universidade de Brasília, UnB, 2017.
- BCG. Disponível em: <<https://www.bcg.com/pt-br/industries/engineered-products-infrastructure/industry-4.0-critical-questions-equipment-suppliers>>. Acesso em 14 set. 2020.
- BEESELEY, P.; CRESS, K. *Architectural Design in Open-Source Software: Developing MeasureIt-ARCH*, an Open Source tool to create Dimensioned and Annotated Architectural drawings within the Blender 3D creation suite. 2019.

BEIRÃO, J.; SANTOR, D. M. *Data collection and constructive classification of superadobe buildings*. Ciência e Sustentabilidade – CeS, 2016.

BHARGAVA, A, Y. *Entendendo Algoritmos: Um guia ilustrado para programadores e outros curiosos*. Ed. Novatec, São Paulo, 2017.

BLENDER. Disponível em: <<https://www.blender.org/>>. Acesso em: 19 set. 2021.

BOSTAN DYNAMICS. Disponível em: <<https://www.bostondynamics.com/>>. Acesso em: 19 set. 2021.

BOS, F.; WOLFS, R.; AHMED, Z.; SALET, T. *Additive manufacturing of concrete in construction: potentials and challenges of 3D concrete printing*, (2016) *Virtual and Physical Prototyping*, 11:3, 209-225, DOI: 10.1080/17452759.2016.1209867

BRAGA, R. C.; NASCIMENTO, D. M. *Déficit habitacional: um problema a ser resolvido ou uma lição a ser aprendida*. Revista de pesquisa em arquitetura e urbanismo. Programa de Pós-Graduação do Departamento de Arquitetura e Urbanismo, v. EESC-USP, 2009.

BRASIL. Federação das Indústrias do Estado de São Paulo (FIESP). *A corrida tecnológica rumo à Indústria 4.0: quem está na pole position?*. In: Cadernos FIESP sobre Indústria 4.0, São Paulo, 2017.

BRASIL. Instituto de estudos para o desenvolvimento industrial (IEDI). ALMEIDA, J. S; CAGNIN, R. F. (Org.). *A indústria do futuro no Brasil e no mundo*. 2019.

CARVALHO, D. R.; DEIRÓ, T. V. *Architectes and refugee camps: a case study for bottom-up approaches*. SIGRaDI22, 2018.

CHAN, K. et al. *The impact of 3D printing technology on the supply chain: Manufacturing and legal perspectives*. International Journal of Production Economics, 2018.

CONFEDERAÇÃO NACIONAL DA INDÚSTRIA. *Investimentos em Indústria 4.0*. Brasília: CNI, 2018.

DAVID, M. J.; KIM, S. H.; XU, M. *The fourth industrial Revolution: Opportunities and Challenges*. International Journal of Financial Research, 2018.

DAVIS, S. R. *Começando a programar em C++ para leigos*. Ed. Altabooks, Rio de Janeiro, 2011.

DEJAN. *How to Mechatronics*, (2021 [s.d.]). Disponível em: <<https://howtomechatronics.com/tutorials/arduino/how-to-control-stepper-motor-with-a4988-driver-and-arduino/>>. Acesso em: 12 jan. 2021.

DFAB House. Disponível em: <<https://dfabhouse.ch/dfab-house/>>. Acesso em: 30 de nov. 2020.

EHANN, S.; WARNIER, C. *Printing Things*. Alemanha: Die Gestalten Verlag, 2014.

FABFAUDANTION. Disponível em: <<https://www.fabfoundation.org/>>. Acesso em: 12 Jul. 2019.

FILAMENT2PRINT. Disponível em: <<https://filament2print.com/pt/bio-sustentaveis/1185-buzzed-pla-cerveza.html>>. Acesso em: 25 set. 2021.

FORBELLONE, A. L. V.; EBERSPACHER, H. F. *Lógica de Programação: A construção de algoritmos e estruturas de dados*. São Paulo: Prentice Hall, 2005.

FLORÊNCIO, E. Q.; SEGUNDO, D. B. F.; QUINTELLA, I. P. C. P. *O Futuro do processo construtivo? A impressão 3D em concreto e seu impacto na concepção e produção da arquitetura*. SiGradi 2016, XX Congreso de la Sociedad Ibero-americana de Gráfica digital 9-11, Novembro, 2016, Buenos Aires, Argentina.

FRATELLO, V. S.; RAEL, R. *Clay bodies: Crafting the future with 3D*. 2017.

FRATELLO, V. S.; RAEL, R. *Innovating materials for large scale additive manufacturing: Salt, soil, cement and chardonnay*. 2020.

FUKUYAMA, M. *Society 5.0: Aiming for a New Human-Centered Society*. Japan SPOTLIGHT, July/August 2018.

GITHUB, 2021. Disponível em: <<https://github.com/slic3r/Slic3r>>. Acesso em: 23 set. 2021.

GNU, [s.d.]. Disponível em: <<http://www.gnu.org/>>. Acesso em: 11 jan. 2021.

GRACE, K. *Primeira comunidade de casas impressas em 3D para desabrigados está sendo construída no México*. Tradução LIBARDONI, V. Disponível em: <<https://www.archdaily.com.br/br/930694/primeira-comunidade-de-casas-impressas-em-3d-para-desabrigados-esta-sendo-construida-no-mexico>> 2019. Acesso em: 21 set. 2021.

HOFFMANN, M. V.; HEISE, A. F.; MINTO, F. C. *Taipa de pilão*. Bauru: FEB-UNESP, PROTERRA, 2011.

HONGXIONG Y., JACKY K. H.; CHUNG, Y.; CHEN; YIJIA, L. *The cost calculation method of construction 3D printing aligned with the internet of things*. Journal on wireless Communications and Networking, 2018.

IEDI. *A Indústria do Futuro no Brasil e no Mundo*. Instituto de Estudos para o Desenvolvimento Industrial. Coletânea, Março, 2019.

INDUSTRIA40, [s.d.]. Disponível em: <<http://www.industria40.gov.br/>>. Acesso em 17 set. 2020.

KESTELIER, X. et al. *Lunar Outpost Design: 3D printing regolith as a construction technique for environmental shield on the moon*. Reserch & Development specialist modelling group, 2009.

KILLA DESIGN. *Escritório do futuro (Office of the future)*/Killa Design. Disponível em: <<https://www.archdaily.com.br/br/877422/escritorio-do-futuro-kill-design>> 2016. Acesso em: 21 set. 2021.

KLEIN, J. et al. *Glass I MIT Media Lab*, 2020. Disponível em: <<https://mediatedmattergroup.com/glass-i>>. Acesso em: 22 nov. 2020.

KONTOVOURKIS, O.; TRYFONOS, G. *Integrating parametric design with robotic additive manufacturing for 3D clay printing: An experimental study*. ISARC, 2018.

KONTOVOURKIS, O.; TRYFONOS, G. *Robotic 3D clay printing of prefabricated non-conventional wall components based on a parametric-integrated design*. Automation in construction, 2020.

KRUGELIS, L. *3D printing technology as a method for discovering new creative opportunities for architecture and design*. Scientific Journal of Latvia University of Life Sciences and Technologies Landscape Architecture and Art, Volume 13, Number 13, 2018.

MARIJNISSEN; MARJOLEIN, P. A. M.; VAN DER ZEE, A. *3D Concrete Printing in Architecture - A research on the potential benefits of 3D Concrete Printing in Architectur*. [s.l.]. 2017.

MIT FABLAB. Disponível em: <<https://fab.cba.mit.edu/>>. Acesso em: 21 set. 2021.

NEVES, C.; FARIA, O. B. *Técnicas de Construção com terra*. Bauru, SP.: FEB-UNESP, PROTERRA, 2011.

NEVES, C.; MILANI, A. P. *Bloco de terra comprimida - BTC*. Bauru, SP.: FEB-UNESP, PROTERRA, 2011.

NUNES, A. *Projeto sustentável*, 2021. Disponível em: <<https://alexnunes.com.br/post.php?evto=107>>. Acesso em: 9 jan. 2021.

OCULUSBY FACEBOOK. Disponível em: <<https://www.oculus.com/>>. Acesso em: 19 set. 2021.

OKUYAMA, F. O.; MILETTO, E., M.; NICOLAO, M. *Desenvolvimento de software I: Conceitos Básico*. Porto Alegre, Bookman, 2014.

PONTE, M. M. C. C. *Arquitetura de Terra: O desenho para a durabilidade das construções*. Dissertação—Faculdade de Ciências e Tecnologia da Universidade de Coimbra: [s.n.], 2012.

PORTAL G1, Distrito Federal. Disponível em: <<https://g1.globo.com/df/distrito-federal/noticia/2021/06/14/estudantes-fazem-protese-de-maos-3d-para-distribuicao-gratuita-no-df.ghtml>>. Acesso em: 13 set. 2021.

POTERWARE., (2020 [s.d.]). Disponível em: <<http://emergingobjects.com/>>. Acesso em: 20 nov. 2020.

PRINTRUN/PRONTERFACE, 2021. Disponível em: <<https://github.com/klimentPrintrun>>. Acesso em: 14 set. 2021.

PYTHON SOFTWARE FOUNDATION. Disponível em: <<https://www.python.org/psf/>>. Acesso em: 22 set. 2021.

REPRAP.ORG. Disponível em: <<https://reprap.org/wiki/G-code>>. Acesso em: 07 out. 2021.

SAMAGAIA, R., NETO, D. D. (2015). *Educação científica informal no movimento maker*. Trabalho apresentado no X Encontro Nacional de Pesquisa em Educação em Ciências, Águas de Lindoia, Brasil. Disponível em: <<http://www.xenpec.com.br/anais2015/resumos/R0211-1.PDF>>. Acesso em: 5 set. 2020.

SANTOS, A. *Projeto no RN Constrói 1ª casa do Brasil com Impressora 3D*. Disponível em: <<https://www.cimentoitambe.com.br/massa-cinzenta/projeto-no-rn-constroio-1a-casa-do-brasil-com-impressora-3d/>> 2020. Acesso em: 14 nov. 2021.

SANTOS, B. P. et al. *Indústria 4.0: desafios e oportunidades*. Artigo, Revista Produção e Desenvolvimento, v.4, n.1, p.111-124, 2018.

SANTOS, E. A. P. *Made in China 2025: passaporte para o estágio intensivo do capitalismo?*. FAU USP, São Paulo, 2018. Disponível em: <https://www.fau.usp.br/docentes/deprojeto/c_deak/AUP840/6t-alun/2018/m2/m2-made-in-china.pdf>. Acesso em 17 ago. 2020.

SCHWAB, K. *A Quarta Revolução Industrial*. Tradução Daniel Moreira Miranda. Edipro, São Paulo, 2016.

SLIC3R. Disponível em: <Manual <https://manual.slic3r.org/intro/overview>>. Acesso em: 23 set 2021.

_____. Disponível em: <<https://slic3r.org/>>. Acesso em: 9 jan. 2021.

SOMMERVILLE, I. *Software Engineering*. Boston, EUA: Pearson Education, 1996.

STOUHI, D. *ETH constrói primeiro projeto de arquitetura em escala real usando uma impressora 3D de areia*. Tradução MARTINO, G. 10 de Agosto de 2018. Disponível em: <https://www.archdaily.com.br/br/899705/eth-constroio-primeiro-projeto-de-arquitetura-em-escala-real-usando-uma-impressora-3d-de-areia?ad_source=search&ad_medium=search_result_all>. Acesso em: 21 set. 2021.

SUPERLICER., 2021. Disponível em: <<https://github.com/supermerill/SuperSlicer>>. Acesso em: 9 jan. 2021.

SUPPLY CHAIN MAGAZINE. Disponível em: <<https://www.supplychainmagazine.pt/2019/05/13/amazon-admite-que-armazens-totamente-automatizados-ainda-estao-longe>>. Acesso em: 20 set. 2021.

VENTAVOLI, F, M, A. *Lógica de Programação com VisualG*. Publicação Digital: E-book. 2014.

3D WASP.,2021. Disponível em: <<https://www.3dwasp.com/en//>>. Acesso em: 14 set. 2021.

3DHOMECONSTRUCTION. Disponível em: <<https://www.instagram.com/3dhomeconstruction/>>. Acesso em 14 nov. 2021.

3DLAB: Soluções em impressão 3D. Disponível em: <<https://3dlab.com.br/tag/tipos-de-preenchimento/>>. Acesso em: 23 out. 2021.

3DMENSIONALS. Disponível em: <<https://www.3dimensionals.de/simplify3d-all-in-one-3d-drucksoftware>>. Acesso em: 06 out. 2021.

3DSLICER., [s.d.]. Disponível em: <<https://discourse.slicer.org/t/fix-a-newly-created-parameter-node/10391>>. Acesso em: 9 jan. 2021.

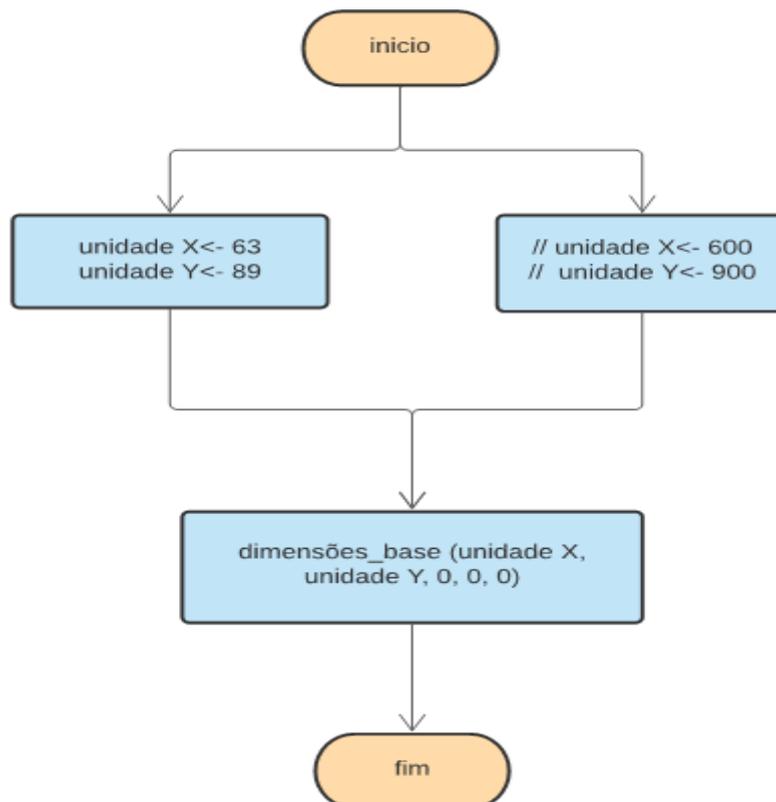
APÊNDICE A – FLUXOGRAMA ALGORITMO BASE_METROS

O estudo apresentou em formato de algoritmos as definições do funcionamento de uma ferramenta digital para impressão tridimensional com argamassas. Os algoritmos apresentados no corpo do texto foram desenvolvidos em “português estruturado”, Portugol. O apêndice disponibiliza os algoritmos em formato de fluxograma que auxiliam no entendimento dos códigos criados e no desenvolvimento futuro de ferramentas digitais com diferentes linguagem de programação:

variaveis unidade X, unidade Y: inteiros; // declaração de variáveis.

grade ← (1 cm; 10cm);

grade ← (10 cm ; 100cm);

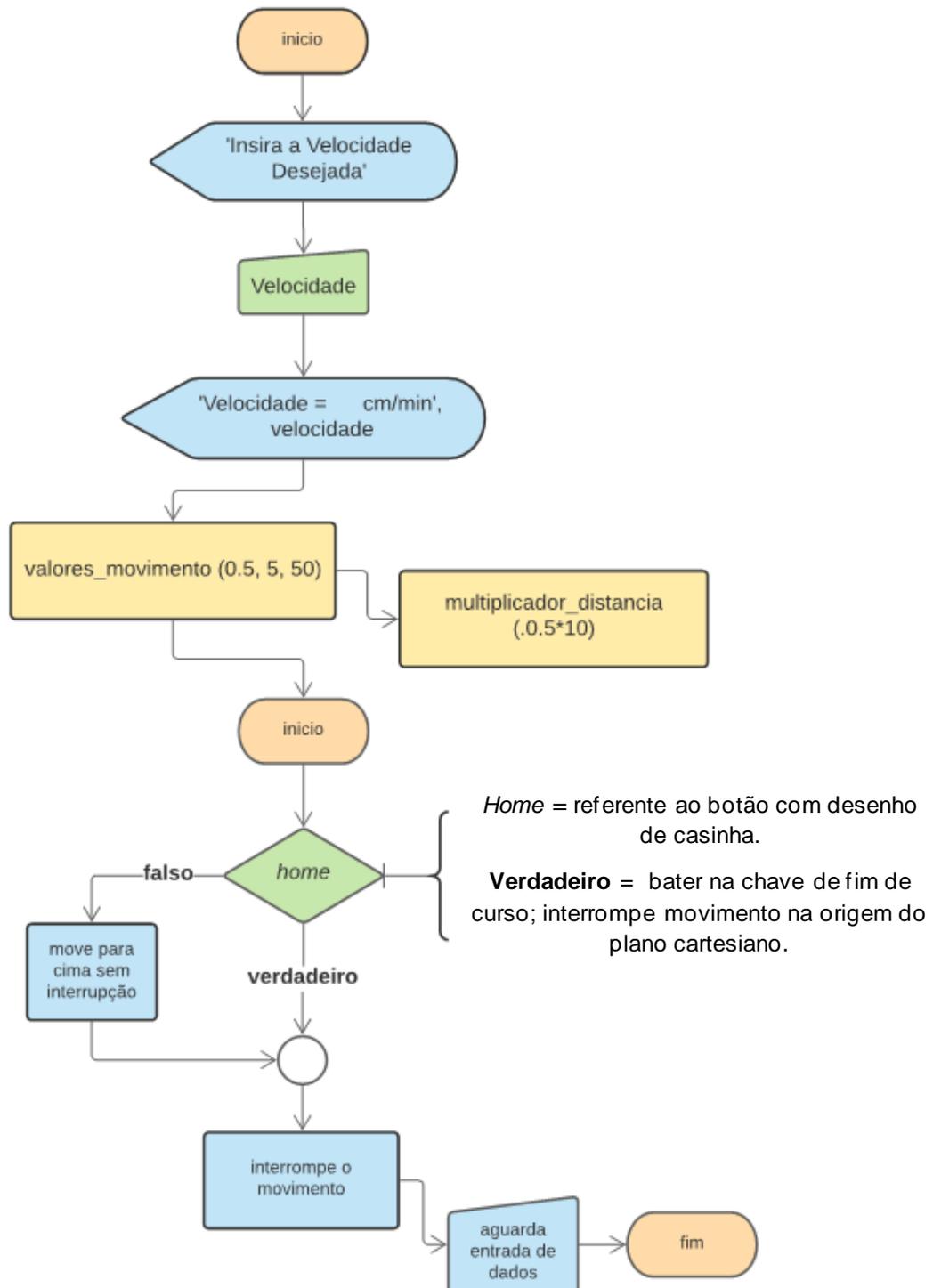


APÊNDICE B – FLUXOGRAMA ALGORITMO MOVIMENTO_Z

variáveis velocidade, valores_movimento: real;

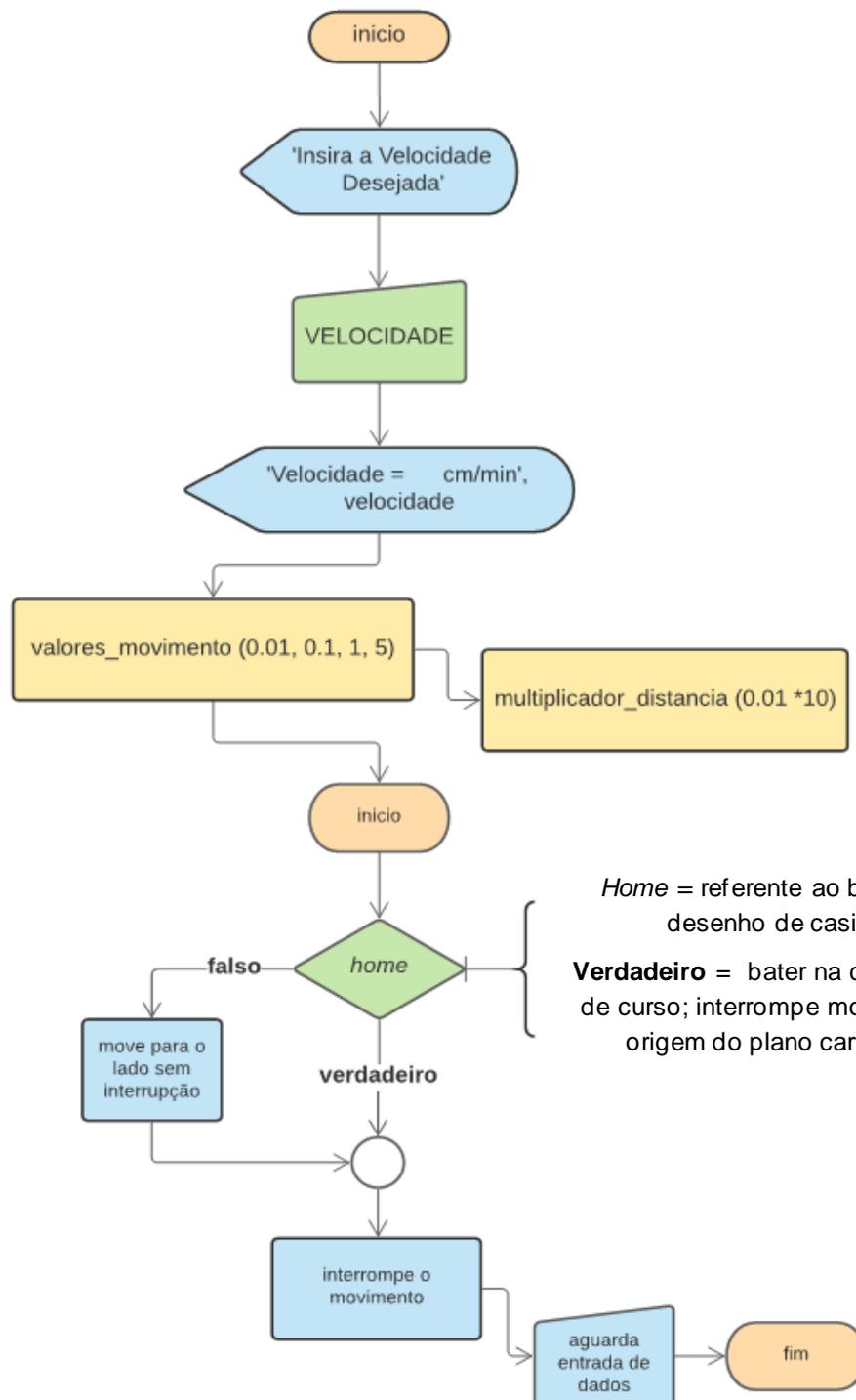
multiplicador_distancia: inteiro;

constante *home*: lógico;



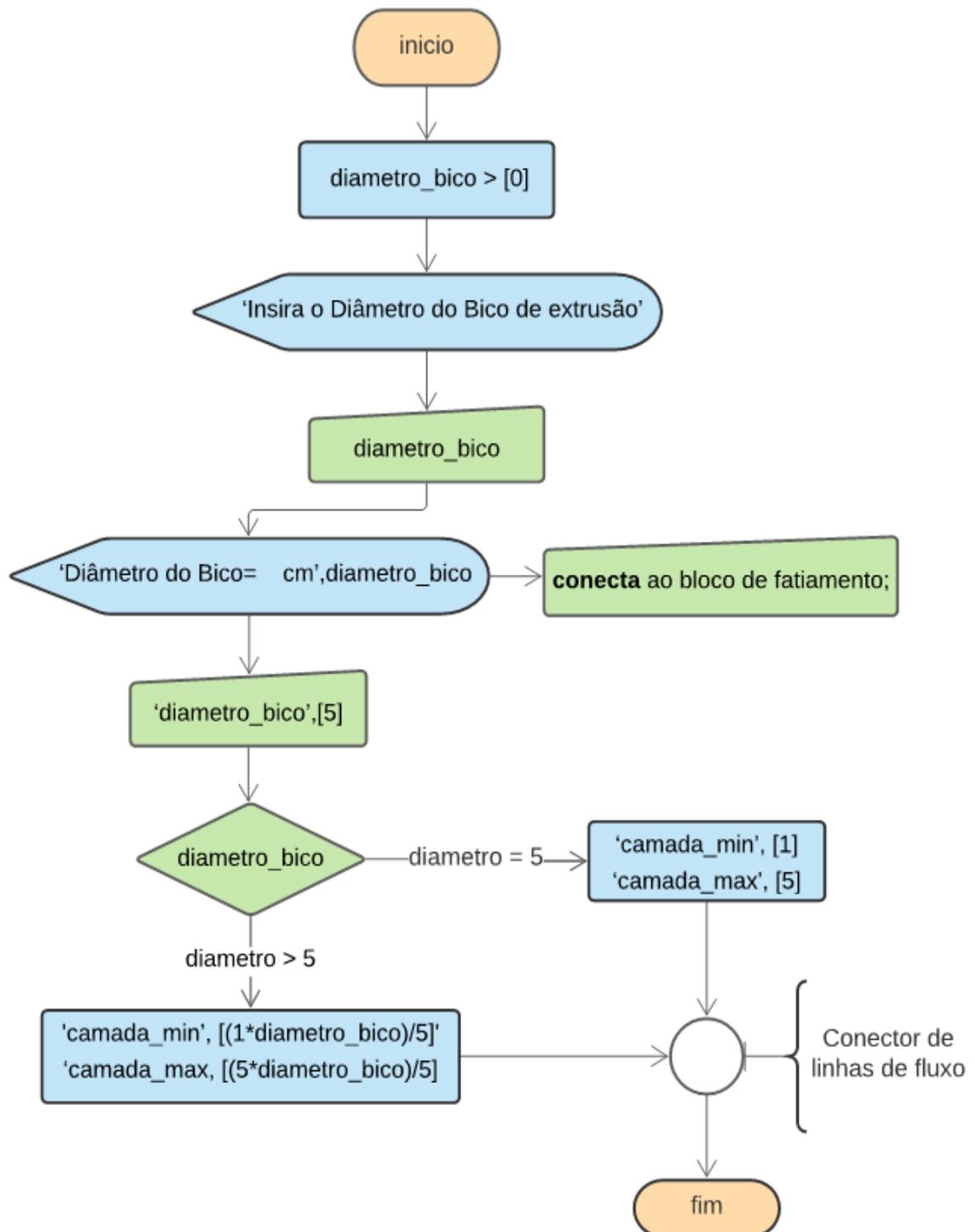
APÊNDICE C – FLUXOGRAMA ALGORITMO MOVIMENTO_XY

variáveis velocidade, valores_movimento: real;
 multiplicador_distancia: inteiro;
 constante *home*: lógico;



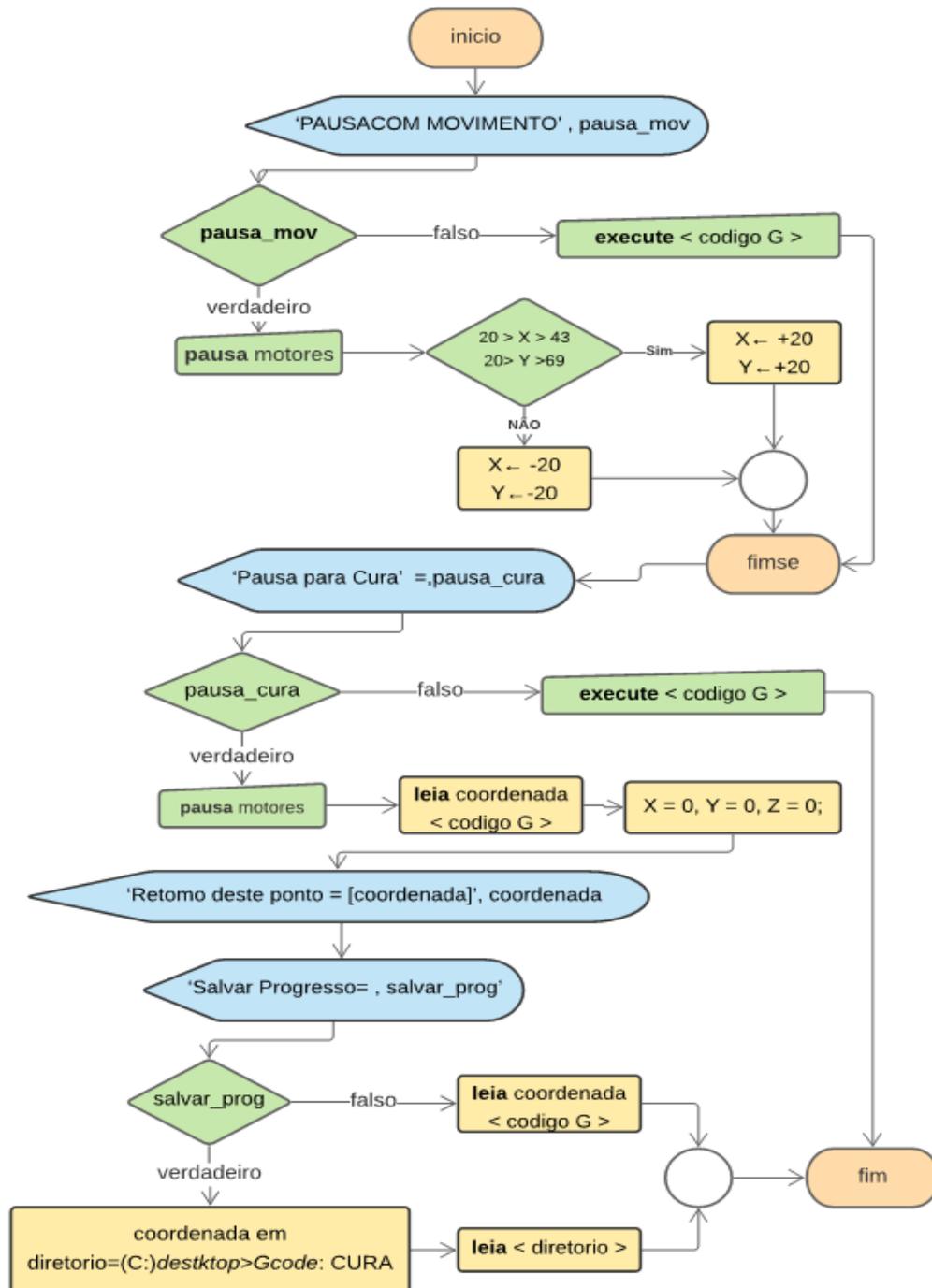
APÊNDICE D – FLUXOGRAMA ALGORITMO BICO_EXTRUSAO

variaveis diametro_bico: inteiro;
camada_min, camada_max: real;



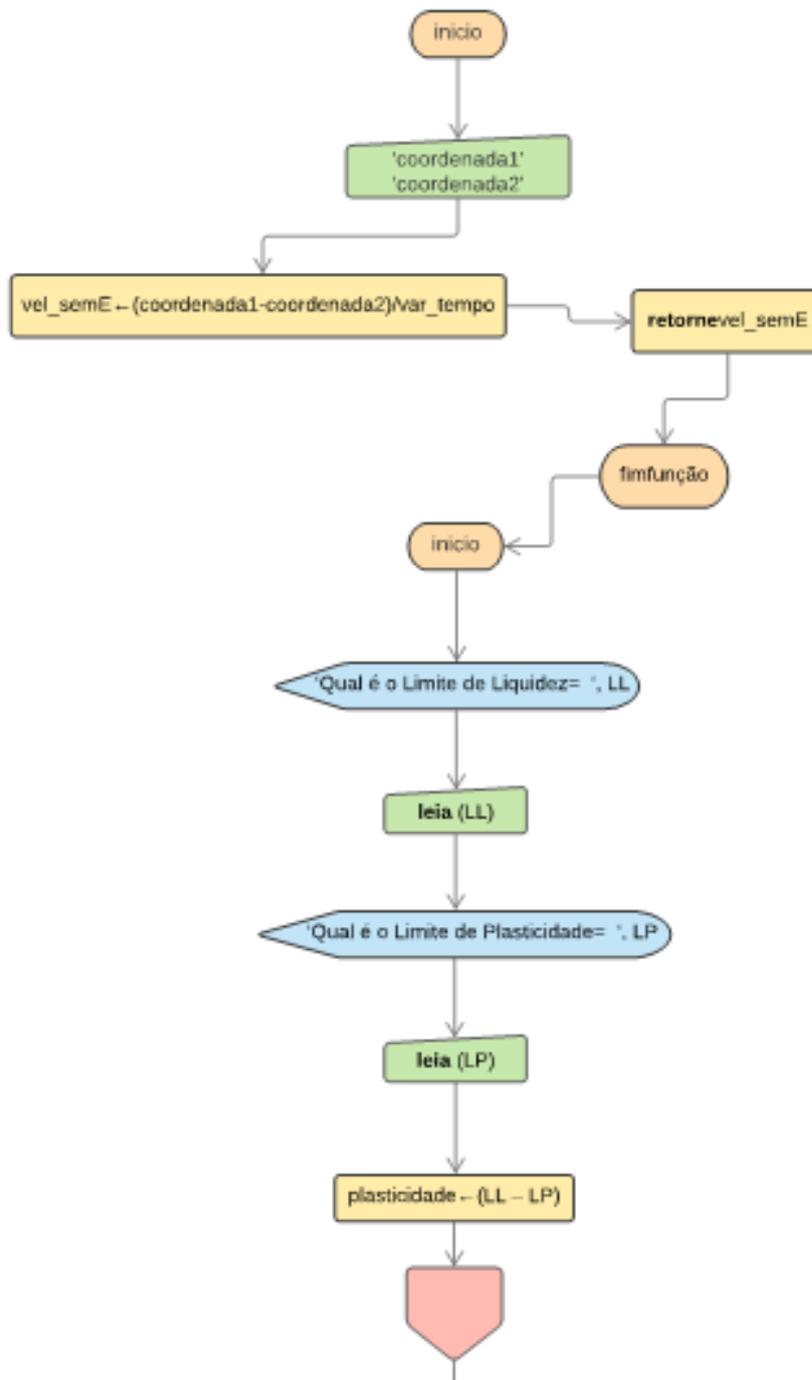
APÊNDICE E – FLUXOGRAMA ALGORITMO PAUSA_RETOMADA

variaveis pausa_mov , pausa_cura, salvar_prog, pausa: lógico;
 coordenada: vetor [4];
 constante diretorio: vetor[3];



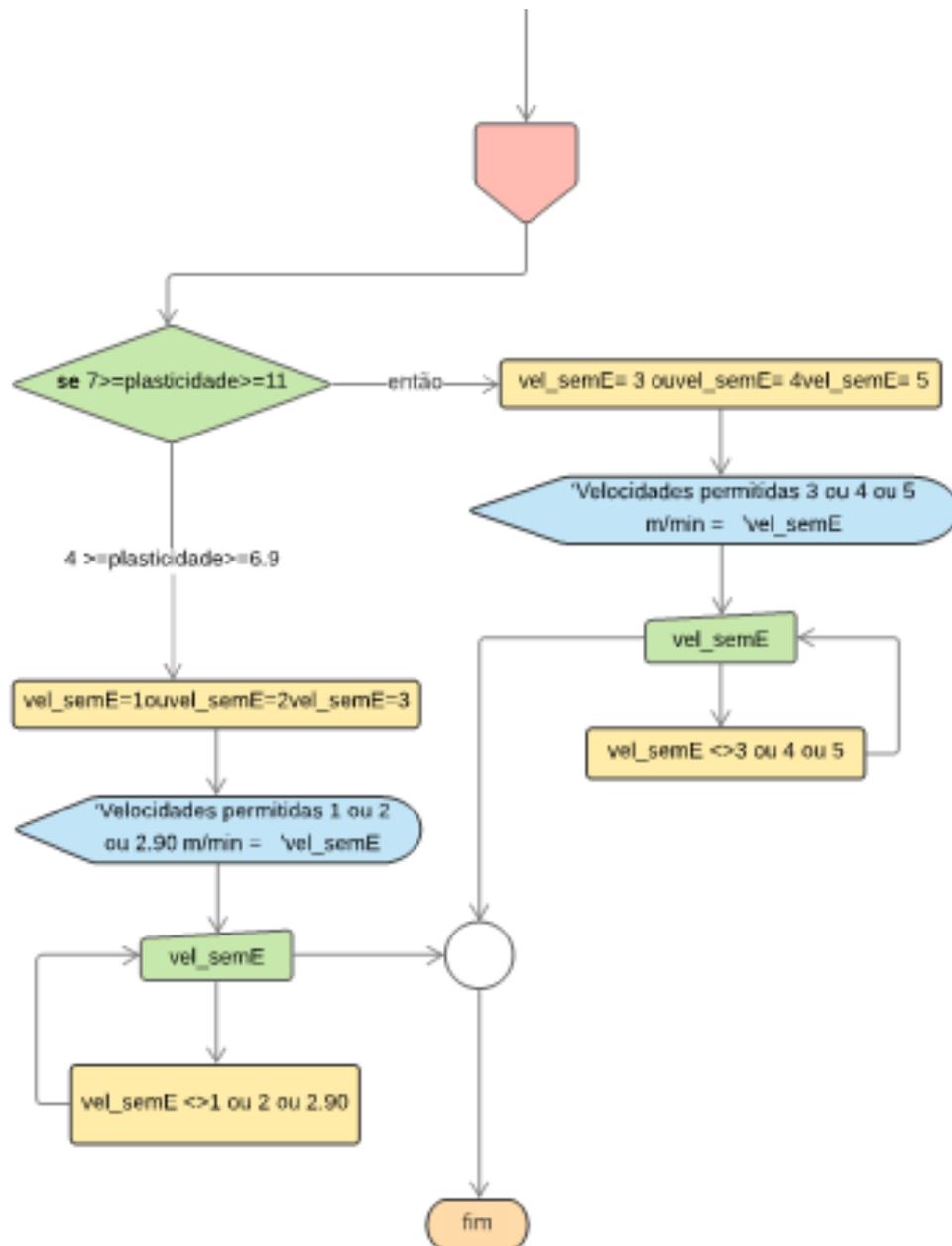
APÊNDICE F – FLUXOGRAMA ALGORITMO VELOCIDADE_SEM_EXTRUSAO

variaveis , coordenada 1, coordenada2, var_tempo,
 vel_semE: real; plasticidade, LL, LP: real;



(continua).

(conclusão).

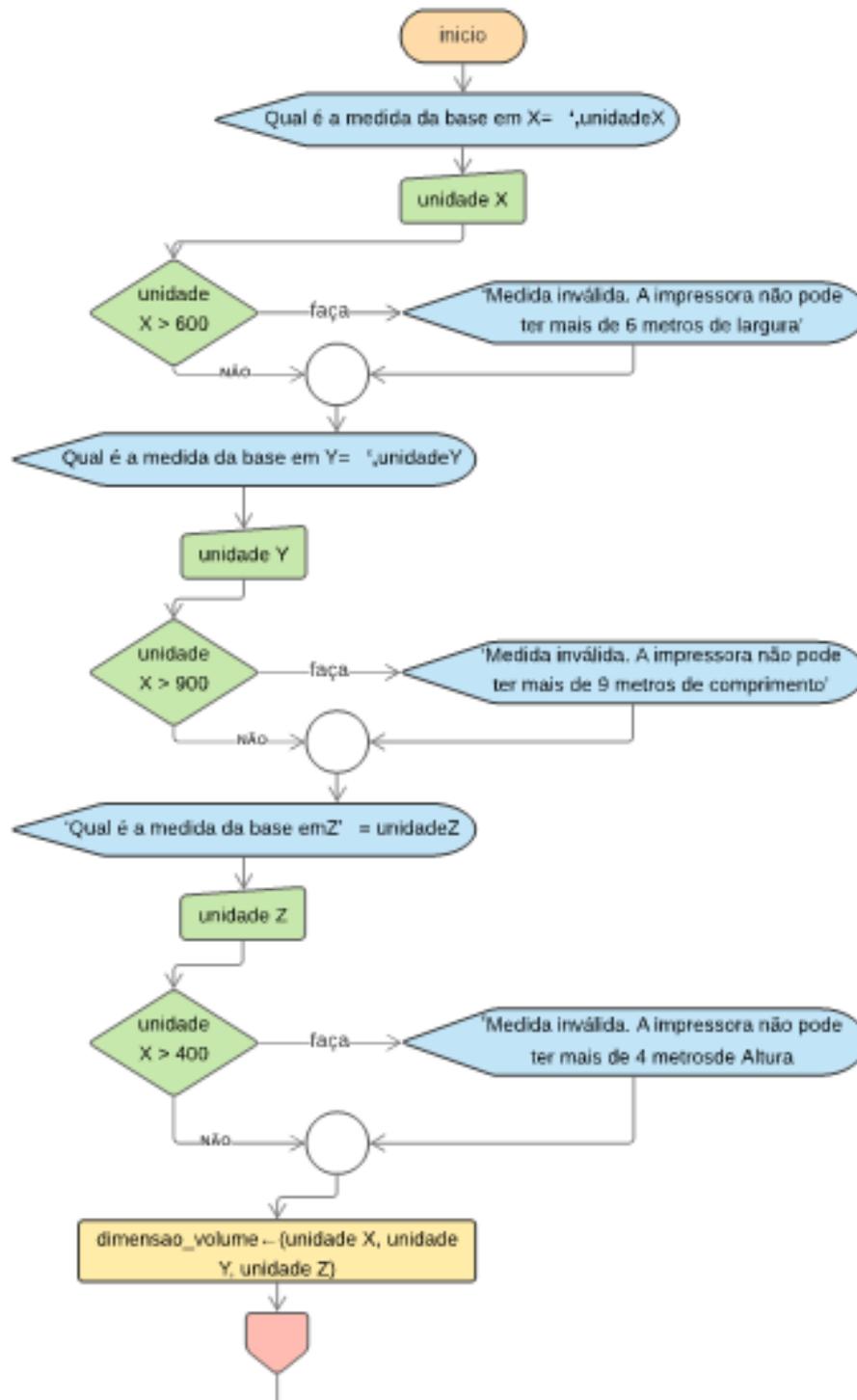


APÊNDICE G – FLUXOGRAMA ALGORITMO VOLUME_METROS

variaveis unidade X, unidade Y, unidade Z: inteiros;

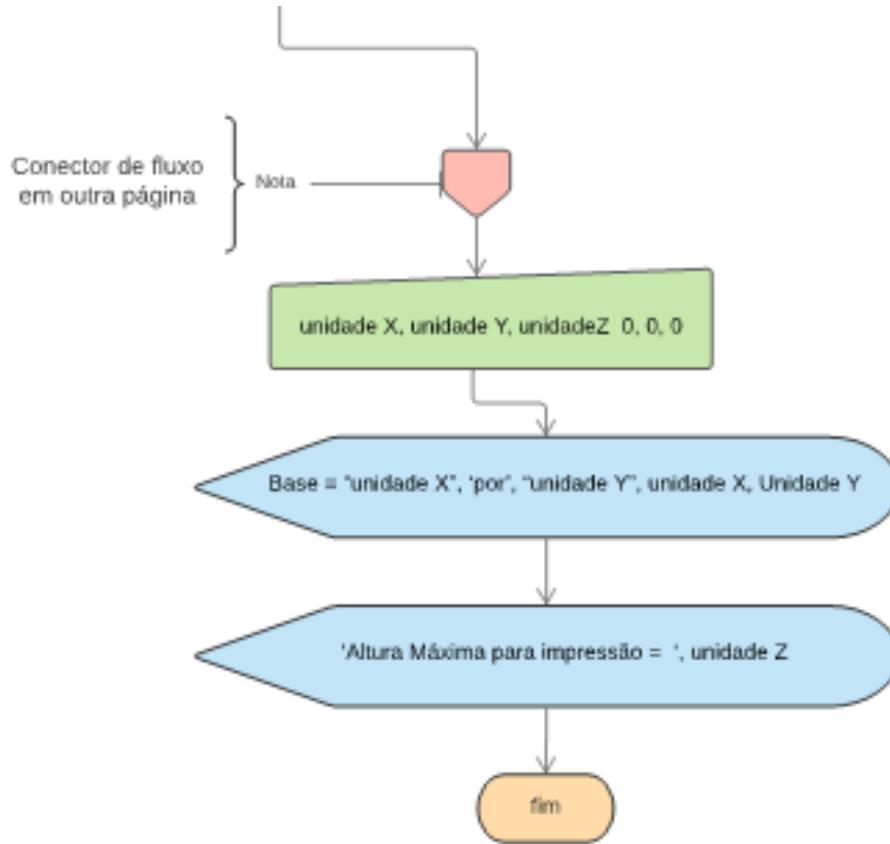
grade ← (10 cm; 100cm);

subgrade ← (100 cm por 100 cm); dimensao_volume: inteiros;



(continua).

(conclusão).

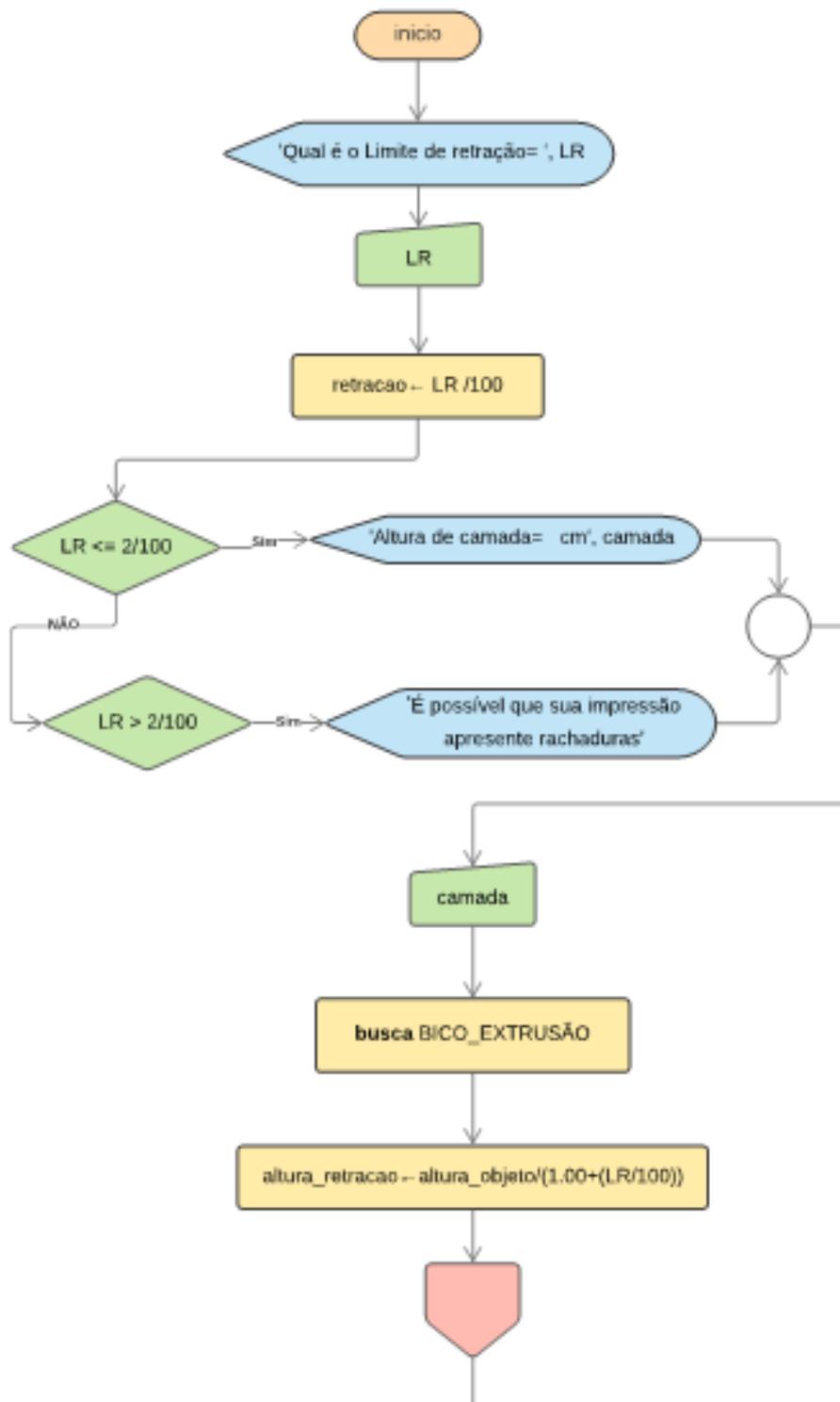


APÊNDICE H – FLUXOGRAMA ALGORITMO RETRAÇÃO

Inclua biblioteca Matematica ← mat;

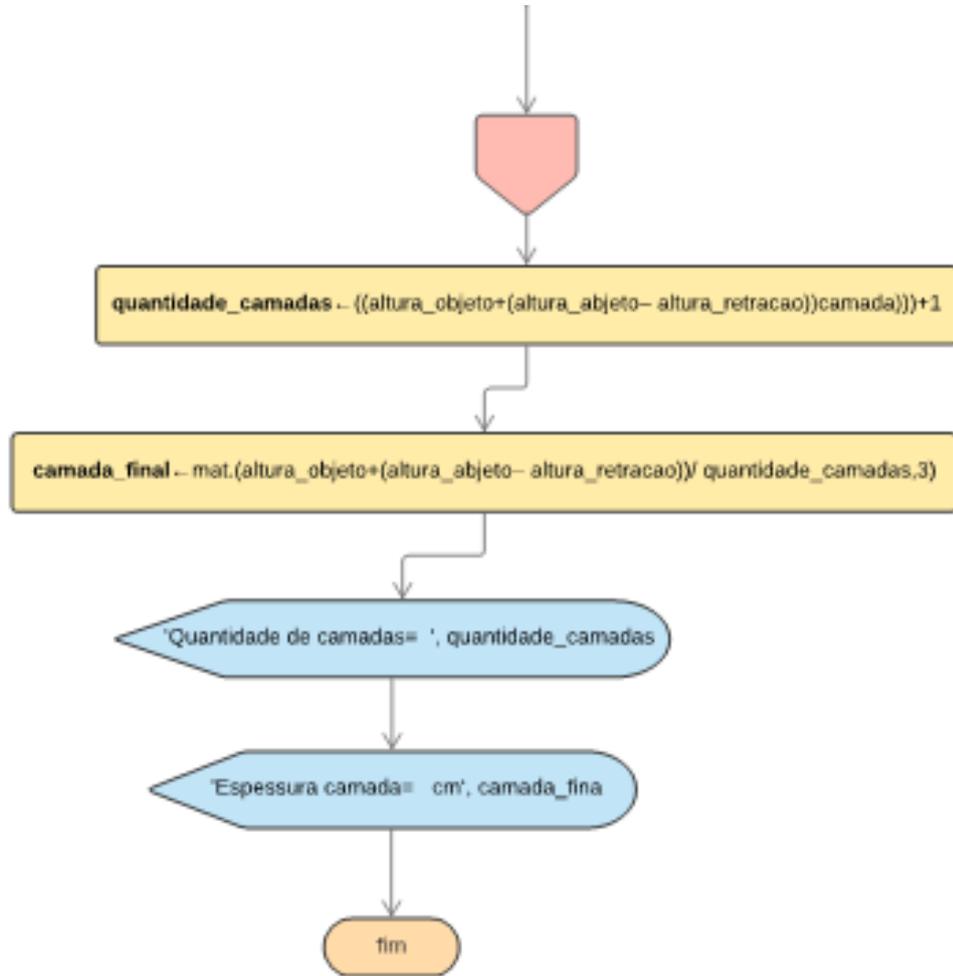
variaveis LR, retracao, camada, altura_objeto: inteiros, camada_final;

altura_retracao, quantidade_camadas: real;



(continua).

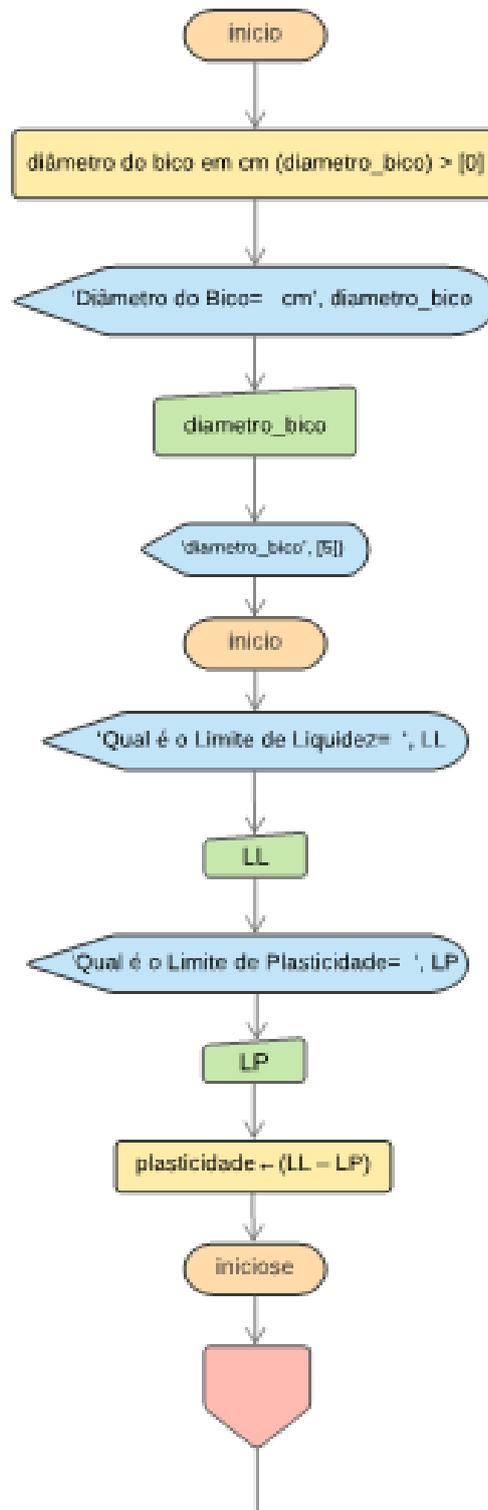
(conclusão).



APÊNDICE I – FLUXOGRAMA ALGORITMO PLASTICIDADE_VELOCIDADE

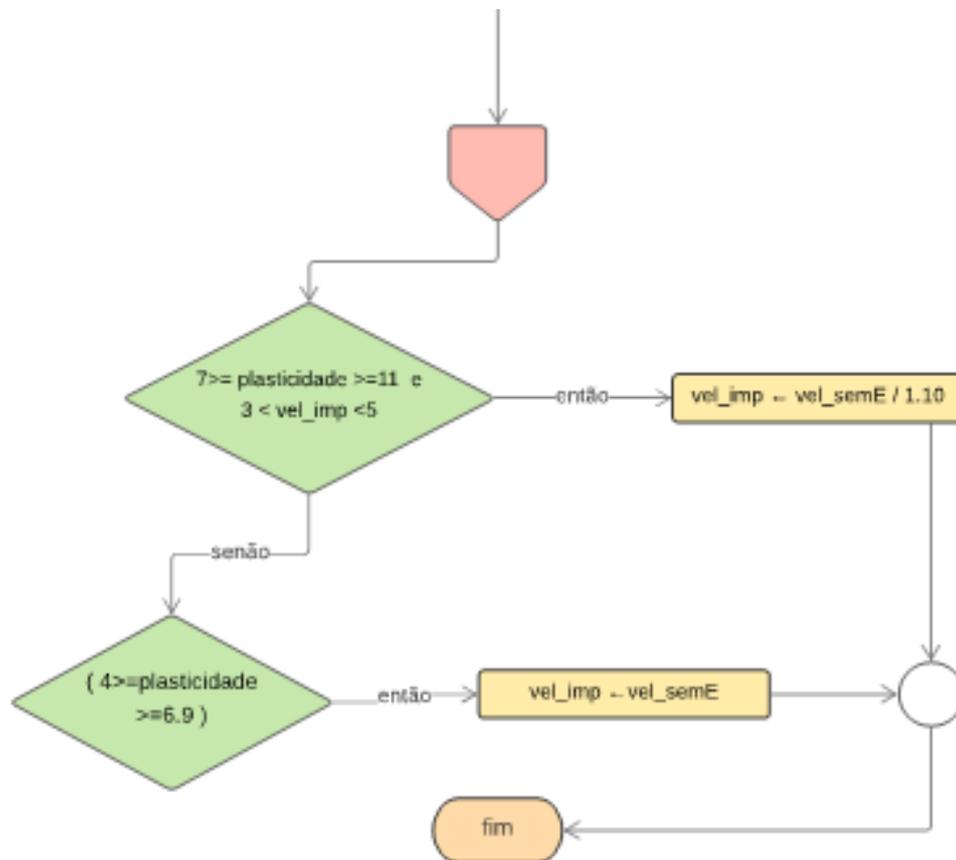
variaveis diametro_bico, vel_imp: inteiro;

plasticidade, LL, LP: real;



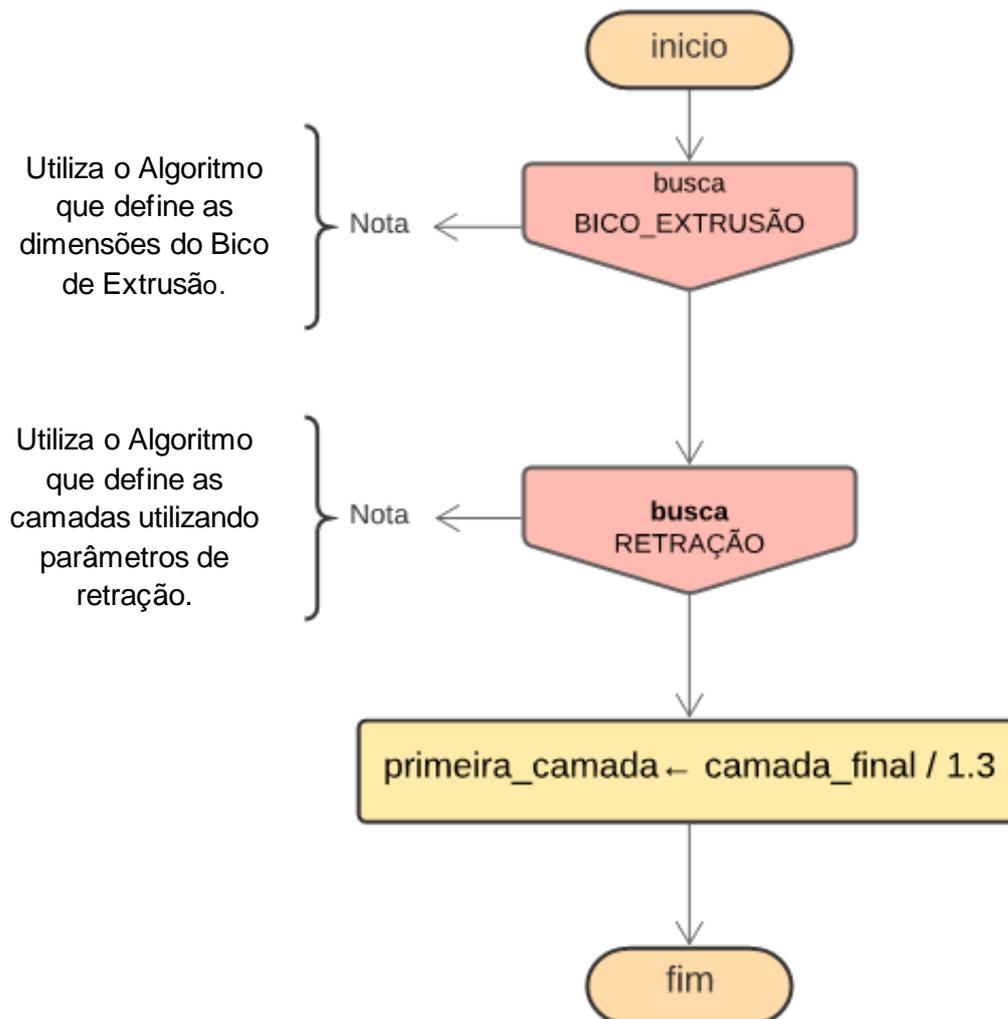
(continua).

(conclusão).



APÊNDICE J – FLUXOGRAMA ALGORITMO PRIMEIRA_CAMADA

variaveis diametro_bico, camada_final, primeira_camada: inteiro;



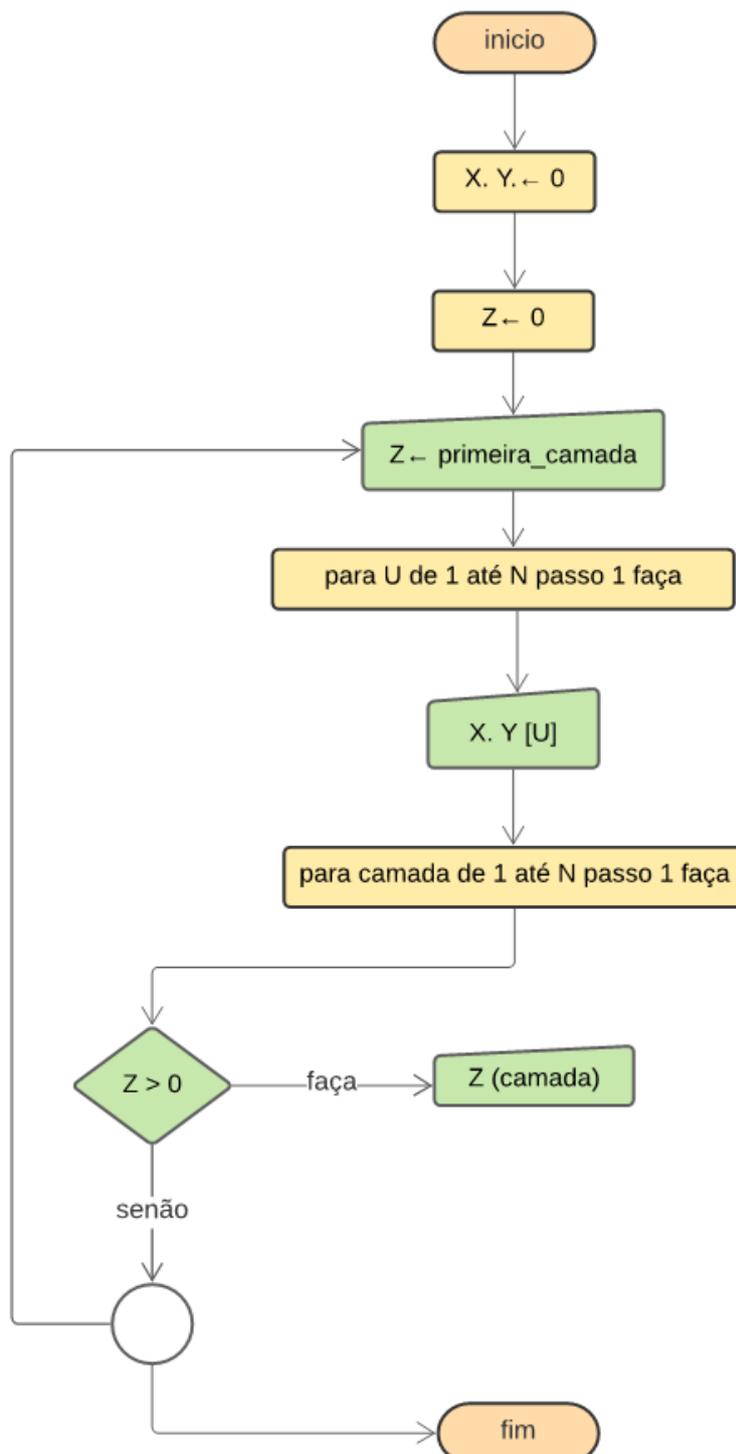
APÊNDICE K – FLUXOGRAMA ALGORITMO ZIGUE_ZAGUE

vetor: $V [1\dots N]$ de inteiros;

variáveis compostas $X.Y, Z$: inteiro;

Constante U : real; // número de coordenadas cartesianas.

camada: inteiro;

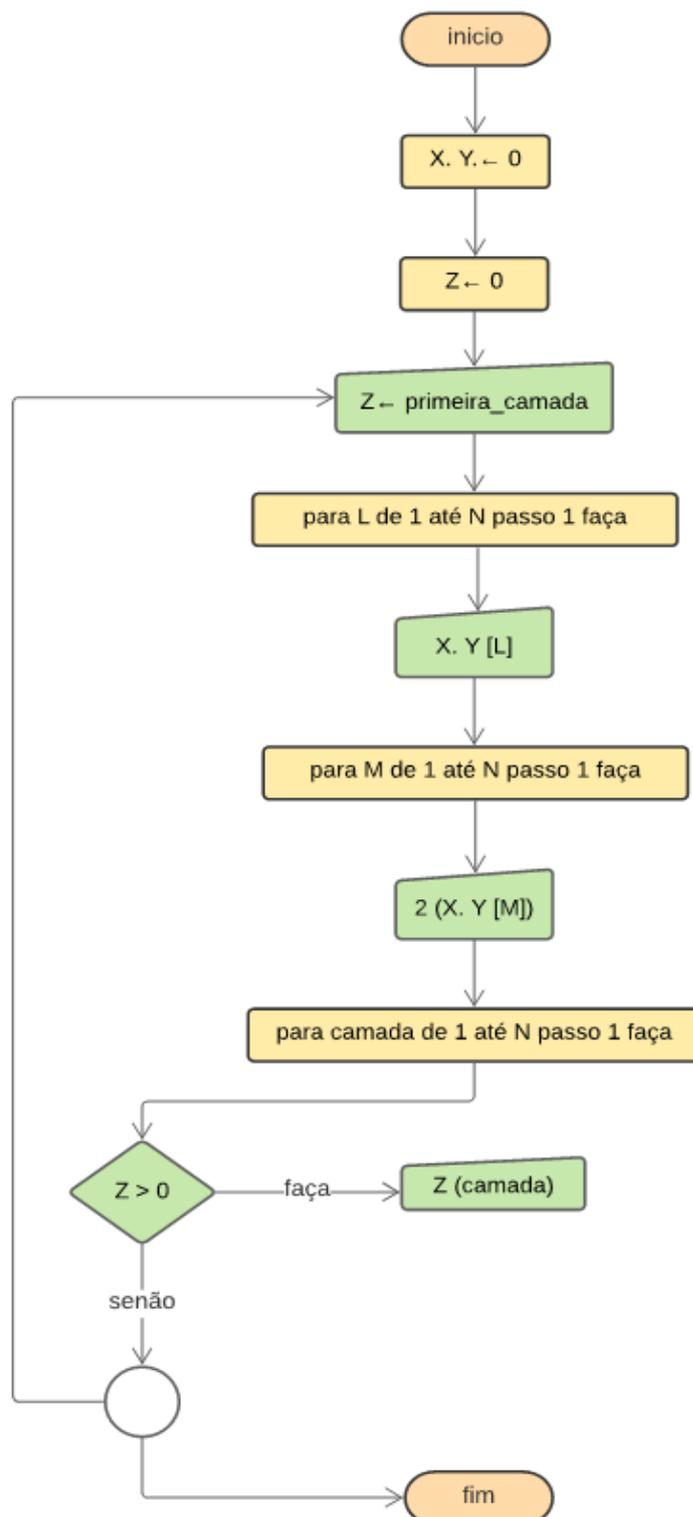


APÊNDICE L – FLUXOGRAMA ALGORITMO GRADE

vetor: $V [1\dots N]$ de inteiros;

variáveis compostas X, Y, Z , : inteiro;

Constante L, M : real;



APÊNDICE M – PROGRAMAÇÃO DA IMPRESSORA 3D DE EXTRUSÃO DE ARGAMASSA – PPG-FAU/UnB

O estudo apresentou uma impressora tridimensional projetada e fabricada na Faculdade de Arquitetura e Urbanismo da Universidade de Brasília, FAU/UnB. A impressora foi definida para ser um modelo que permita a simulação de movimentação e impressões de elementos arquitetônicos que poderiam ser produzidos por impressoras tridimensionais de tamanho natural. A produção da impressora permitiu um melhor entendimento da programação e sua modificação para definir a movimentação necessária de acordo com os parâmetros de impressão estipulados no trabalho.

Abaixo, são apresentadas as principais linhas de código da programação da impressora tridimensional da FAU/UnB e como foram modificadas:

Linhas de código modificadas:

89 #define STRING_CONFIG_H_AUTHOR "RENAN.BALZANI"

// Indica quem é o programador responsável pela modificação

140 #define CUSTOM_MACHINE_NAME "Pos-UnB.08.06" (cf. Figura 75)

// Modifica e apresenta o nome que aparece na tela de principal do visor LCD utilizado na impressora.

Figura 74 – Programação feita para impressora construída nos laboratórios EdiQuali/LFDC da FAU/UnB

```

PASTER_Ub_BUSON.RENAN_rev.23.12 - Configuration.h | Arduino 1.8.16 (Windows Store 1.8.51.0)
Arquivo Editar Sketch Ferramentas Ajuda

// Enable the Bluetooth serial interface on AT90USB devices - DESCOMENTAR O COMANDO SE QUISER UTILIZAR ALGUM SISTEMA DE COMUNICAÇÃO BLUETOOTH
// #define BLUETOOTH

// The following define selects which electronics board you have.
// Please choose the name from Boards.h that matches your setup - TIPO DE PLACA MÊ - PARA IMPRESSORA DE TERRA DO LAB. EDIQUALI ESTAMOS USANDO UMA RAMPs 1.4 - SE PRECISAR MUDAR A PLACA
// PLACA Vá NA 2da Boards.h - PROCURE PELA PLACA NÃO UTILIZADA NO HARDWARE DA IMPRESSORA - VERIFIQUE A VERSÃO DO MARLIN E SE TEM A PLACA ADQUIRIDA.
#ifdef MOTHERBOARD
  #define MOTHERBOARD BOARD_RAMPs_14_EFB
#endif

// Optional custom name for your RepStrap or other custom machine
// Displayed in the LCD "Ready" message - NOME DA IMPRESSORA QUE FICA APARECENDO NA TELA PRINCIPAL DE USO.
#define CUSTOM_MACHINE_NAME "Pos-UnB.08.06" // MODIFICADO PARA A IMPRESSORA DE ARGAMASSA DE TERRA PROJETADA E CONSTRUÍDA NA POS UNB

// Define this to set a unique identifier for this printer. (Used by some programs to differentiate between machines)
// You can use an online service to generate a random UUID. (eg http://www.uuidgenerator.net/version4)
// #define MACHINE_UID "00000000-0000-0000-0000-000000000000" - PARA DIFERENCIAR A IMPRESSORA CASO A MAQUINA FAÇA PARTE DE UMA "FARM" DE IMPRESSÃO - COMO O ESTUDO É PARA IMPRESSORAS EM
// GRANDE FORMATO NÃO É NECESSÁRIO MUDAR ESSE PARÂMETRO.

// This defines the number of extruders - DEFINIR A QUANTIDADE DE EXTRUSORES - A MAQUINA TEM APENAS UM 1 EXTRUSOR.
// :{1, 2, 3, 4}
#define EXTRUDERS 1

// Enable if your E steppers or extruder gear ratios are not identical - ATIVE SE SEUS PASSOS "Z" OU RELAÇÃO DE ENGRENAGEM DE EXTRUSORA NÃO FOREM IDENTICS.
// #define DISTINCT_E_FACTORS

// For Cyclops or any "multi-extruder" that shares a single nozzle.

```

Fonte: o autor.

255 #define TEMP_SENSOR_0 998

// Indica o tipo de sensor utilizado nas impressoras 3D de plástico. A impressora de argamassa não aquece o sistema de extrusão. É utilizado comando 998, que estabelece uma temperatura fictícia de 25°C.

259 #define TEMP_SENSOR_BED 0

// Define o aquecimento da mesa. A impressora de argamassa não possui mesa de deposição aquecida. A temperatura é estabelecida em zero para não gerar conflito.

439 // #define USE_ZMIN_PLUG

// Na linha de comando 439 foram adicionadas duas barras para desabilitar sua função. A impressora foi projetada para utilizar o eixo Z zerando na chave de fim de curso no lado positivo do movimento.

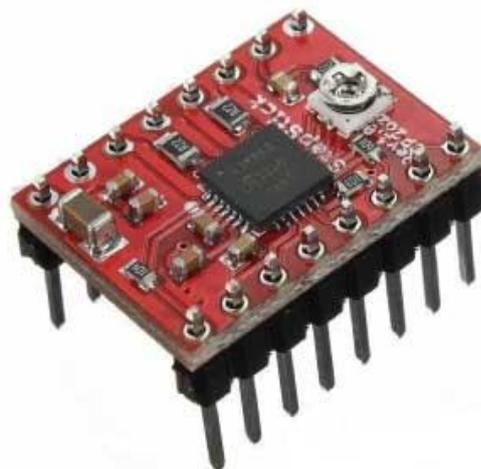
442 #define USE_ZMAX_PLUG

// A linha de código foi “descomentada”, ou seja, habilitada para que a chave de fim de curso do eixo Z fosse colocada no lado positivo do movimento.

491 #define DEFAULT_AXIS_STEPS_PER_UNIT { 80, 80, 400, 1000 }

// ANTES { 80, 80, 2560, 508 }. Essa linha de código define a quantidade de passos/mm e define o quanto cada eixo se movimenta. É importante verificar qual é o *driver* (peça conectada à placa-mãe para fazer o controle fino dos motores de passo). A impressora utiliza *drivers* A4988 (*vide* Figura 75):

Figura 75 – Microcontroladora A4988 para motor de passo bipolar



Fonte: Dejan, 2021.

A impressora utiliza para a movimentação dos eixos motores de passo. Os motores são bipolares, podem girar no sentido horário ou anti-horário e cada passo corresponde a $1,8^\circ$ grau. A torque é transmitida para os eixos, gerando o movimento linear por correias de 2 mm de largura e engrenagens de alumínio com 20 dentes.

O movimento do eixo Z é feito por fusos trapezoidais com 8 mm de largura. Uma revolução completa do fuso corresponde a um movimento vertical de 8 mm para cima ou para baixo.

724 #define Z_HOME_DIR 1

// A programação define o ponto zero dos eixos por padrão como menos um (-1). A máquina foi projetada para zerar o eixo Z no ponto positivo para evitar colisões da ponta do extrusor com o objeto impresso em possíveis pausas ou reinícios de impressão. A impressora tem sua origem ou fim de movimento na parte superior do pórtico de alumínio (cf. Figura 76):

Figura 76 – Eixos de movimento da impressora 3D. Eixo Z tem sua origem/fim de movimento na parte superior do pórtico



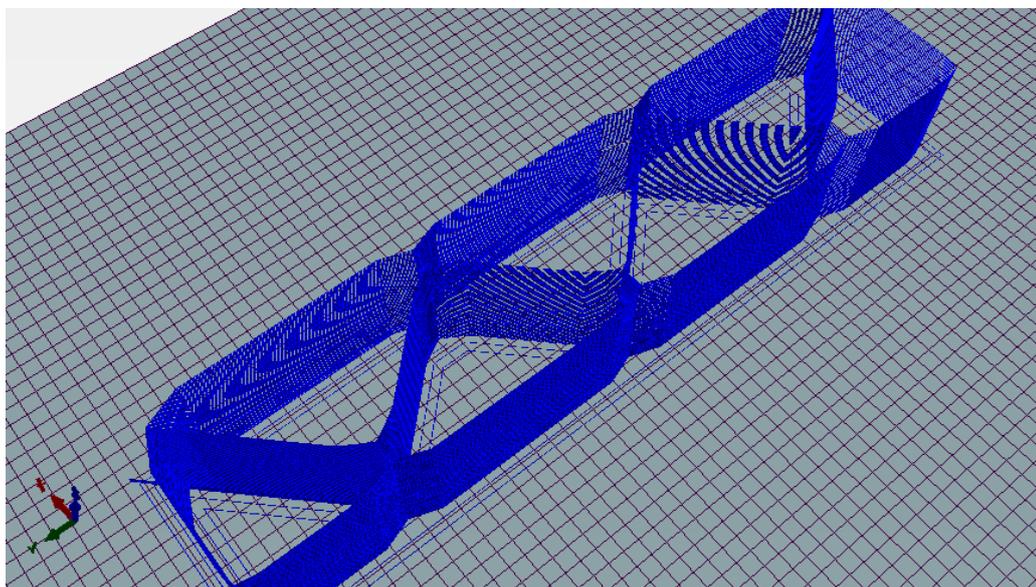
```
733 #define X_MIN_POS 0
734 #define Y_MIN_POS 0
735 #define Z_MIN_POS 0
736 #define X_MAX_POS 630
737 #define Y_MAX_POS 890
738 #define Z_MAX_POS 415
```

// As linhas de código de 733 a 738 definem o tamanho da área de impressão e altura do eixo Z. As medidas estão em milímetros e indicam o movimento máximo dos eixos.

APÊNDICE N – EXEMPLO CÓDIGO G COMENTADO

O estudo apresenta uma simulação para definição de um elemento arquitetônico. O modelo é uma parede de 25 cm de largura, 16 cm de altura e 100 cm de comprimento. A simulação define em modelo virtual em escala 1:2 para apresentar a complexidade de um código G necessário para uma impressão tridimensional. O modelo apresenta o preenchimento em zigue-zague, cf. Figura 77:

Figura 77 – Simulação de impressão de parede com preenchimento em zigue-zague



Fonte: o autor.

O código G criado pela ferramenta apresenta uma grande quantidade de linhas de código. A apresentação da simulação necessita de pelo menos 583.066 linhas. O código abaixo apresenta apenas algumas linhas para exemplificar as principais funções utilizadas em uma impressão 3D com argamassa.

[...]

10 **G28** // comando **G28** zera os eixos permitindo o início da impressão 3D.

11 **G1 Z415 F5000** // Z415 faz com que a impressora da FAU/UnB dessa o eixo e encoste o extrusor na base de impressão. G1 – Movimento linear.

12 **G1 Z410 F5000** // Z410 faz com que o eixo Z suba 5 milímetros para começar
a 21 **G1 X 259.024 Y 731.859 F18000.000** // **G1** é o movimento linear ao movimento dos eixos X e Y definidos pelas coordenadas cartesianas.

22 **G1 F9000** // define com **G1** a velocidade das movimentações lineares com extrusão de argamassa.

23 **G1 X344.872 Y731.859 E175.63634**

24 **G1 X261.524 Y815.221 E416.81228**

25 **G1 X336.517 Y890.215 E633.79537**

26 **G1 X261.525 Y965.220 E850.79404**

// Movimentos lineares no plano X Y. **E** = significa quantidade de material extrudado. A primeira camada deve ser menos espessa para ajudar a fixar a argamassa à base de impressão ou ao solo.

[...]

16 **G21** // define o funcionamento da máquina e a programação em sistema internacional (sistema métrico).

17 **G90** // posicionamento de coordenadas absolutas. O posicionamento absoluto define que a impressora deve se movimentar para uma coordenada XYZ exata do plano cartesiano.

18 **M82** // usa medidas absolutas para extrusão da argamassa.

19 **G92 E0** // o comando G92 define posição dos eixos para que o extrusor seja movimentado para outro ponto no plano cartesiano. O comando **E0** interrompe a extrusão de argamassa no momento do movimento do extrusor.

20 **G1 Z 4.000 F18000.000** // **G1** é o movimento linear associado ao comando **Z 4.000** define que o eixo Z suba 4 milímetros.

[...]

33 **G92 E0** // movimento sem extrusão.

[...]

177 **G1 X245.371 Y1212.215 F18000.000**

178 **G1 F9000**

179 **G1 X245.371 Y718.225 E1010.66168**

180 **G1 X364.371 Y718.225 E1254.12563**

// Conjunto de movimentos no plano XY com extrusão.

192 **G1 Z4.000 F18000.000** // levanta o eixo Z para começar mais uma camada.

[...]

583064 **G28 Z0** // comando G28 zera o eixo Z primeiro para evitar colisões com os elementos arquitetônicos impressões.

583065 **G28 X0 Y0** // comando G28 zera o eixo X para liberar a movimentação do pórtico do eixo Y.

// Os últimos 3 comandos do código G são importantes para definir uma utilização segura das impressoras 3D por usuários menos experientes.

583065 **M84** // desliga os motores da impressora automaticamente. Códigos M referentes à preparação de funções preestabelecidas para máquinas CNC.

Foi comentada apenas uma parcela do código G, pois a programação é um processo repetitivo. O exemplo apresentado é uma parcela do que seria uma impressão e seu código G possui mais de 580.000 linhas de código. Contudo, não é um arquivo pesado, podendo ser salvo com facilidade em cartões de memória.