ADDITIVE MANUFACTURING METHODOLOGIES FOR MULTI PROCESS AND MULTI MATERIAL SCENARIOS

by

TUSHAR SAINI

Presented to the Faculty of the Graduate School of

The University of Texas at Arlington in Partial Fulfillment

of the Requirements

for the Degree of

MASTER OF SCIENCE IN MECHANICAL ENGINEERING

THE UNIVERSITY OF TEXAS AT ARLINGTON

December 2015

Acknowledgements

I would like to take this opportunity to thank my supervising professor Dr. Panos S. Shiakolas for his constant encouragement, support and guidance during the course of my research and studies at this University. The invaluable advice and support provided by him was the major driving force, which enabled me to complete my thesis.

I would like to thank Dr. Pranesh Aswath and Dr. Tre' Welch for taking time to serve on my thesis committee. Also, I would like to thank Prashanth Ravi for his invaluable support and timely inputs.

I would like to thank all my friends at the MARS Lab and in the University for helping me throughout my time here at this University. Finally, I would like to thank my parents for all their support, without which I would not have been able to complete the research.

December 14, 2015

Abstract

ADDITIVE MANUFACTURING METHODOLOGIES FOR
MULTI PROCESS MATERIAL SCENARIOS

Tushar Saini, MS

The University of Texas at Arlington, 2015

Supervising Professor: Panos S. Shiakolas

Additive Manufacturing processes have been developing and continuously improving but are primarily constrained to single materials, albeit some commercial multi extrusion 3D printers do exist where by it is possible to print with multi-colored fluidic polymers or at most two filaments of the same type. 3D printing an object with volumetric varying properties requires manual modification of the G-Code, not a feasible option considering the average size of G-Code generated by the slicing software. This research focuses on developing a post processing methodology that modifies the large G-Code files generated by the slicing software according to user requirements. The proposed methodology has been tested on the Custom Multi-Modality 3D BioPrinter (CMMB) that combines multiple fabrication technologies on a single additive manufacturing platform.

Table of Contents

List of Illustrations

List of Tables

Chapter 1

Introduction

Additive manufacturing (AM) combines a number of processes and technologies that offer a full spectrum of capabilities to manufacture components drawn in a three dimensional (3D) solid modeling software with customized and defined materials and properties. This manufacturing technique uses layer by layer addition of material to create three dimensional components as needed. Most hobby-grade AM machines are based on the open-source self-replicating 3D printer, RepRap. RepRap uses fused-filament fabrication to build engineering components primarily from a variety of thermoplastic materials such as polylactic acid (PLA) and acrylonitrile butadiene styrene (ABS). The current common AM technologies are capable of processing a single material or maximum up to two similar materials on 3D printers such as the Makerbot Replicator 2X, but they are all restricted to a single extrusion process. Some AM machines can use more than two materials using a single process based on polyjet printing like the Stratasys Objet Connex 3D Printer series, which can print with as many as fourteen liquid materials. Yet an area where most of these machines lack is the ability to create functional structures with heterogeneous properties and materials.

The motivation of the current research is to evolve AM using the open source tools available so that the developed methodologies are applicable to most AM machines. The initial aim is to create a structure consisting of two components each created with its own extruder simultaneously to evaluate the general outline and to prove that the multiple extrusion using a single process is indeed possible. This is performed using a simple post processing script which assigns extruders as needed. Structures with mechanically heterogeneous components can be manufactured using AM to create structures and assemblies with desired and varying properties , such as is the case with biomedical scaffolds. Heterogeneous components are designed using design level approach which defines features at design level using unit cells. A post-processing script was also developed that combines G-codes of multiple homogenous constructs at defined heights to create heterogeneous components. It is possible to create structures through in-situ combination of multiple distinctly processed materials and using different processes to enable the manufacturing of structures with desired and customized structural and functional properties.

1.1 Additive Manufacturing

Additive Manufacturing (AM) is a process based on the principle of creating or fabricating components or assemblies in a layer-by-layer addition of material. It is also referred to as Layered Manufacturing (LM) or Rapid Prototyping (RP), though there does not exist a universally accepted terminology. AM addresses two major challenges faced by the manufacturing industry: (1) substantial reduction in product development time; and (2) improvement on flexibility for manufacturing products so as to create a variety of types of products [1]. In order to improve traditional production design and manufacturing, Computer-Aided Design and Manufacturing (CAD and CAM) software tools have been used. However, true integration of computer-aided design with manufacturing for rapid prototyping is still not complete -----and there are gaps between CAD and CAM which remain unfilled in the following aspects: (1) rapid creation of 3D models and prototypes at the industrial level and (2) cost-effective production using multiple materials and multiple processes.

Over the years, various additive manufacturing technologies have emerged including, Stereolithography, Selective Laser Sintering (SLS) Fused Deposition Modeling (FDM), Laminated Object Manufacturing (LOM), Ballistic Particle Manufacturing (BPM), and Three Dimensional Printing (3DP). All these technologies have a common feature that the structure is created by adding material, layer-by-layer, instead of removing material and are capable of directly creating three dimensional structures from a CAD model. This simplifies manufacturing of 3D objects by arranging multiple 2D layers (with predefined thickness or 2½D) one over another. Figure 1-1 shows a CAD model of a teacup that is sliced into layers. The resolution of each layers governs the quality of the final print. It is possible to fabricate components with finer features by using lower layer height.

A typical additive manufacturing process involves several steps whose complexity and more detailed elements depend on the chosen process and the complexity of the three-dimensional object [1].

Figure 1-1 CAD image of a teacup with further images showing the effects of building using different layer thickness [2]

In contrast to other manufacturing techniques AM does not require large amounts of instructions. Where other manufacturing processes, such as, laser sintering require a careful and detailed analysis of the structure geometry to determine features, tools and the order of creation of features using those tools, AM needs only some basic dimensional details and an understanding of how the AM process and corresponding machine operates along with the material(s) used. The key to how AM works is that components are fabricated by controllably depositing material(s) in a layer where each layer is a cross-section of the component. The created object is an approximation dimensionally of the original CAD model since each layer needs to have a defined thickness associated with it. It is expected that the thinner the layer, the more accurate the manufactured object, however one needs to consider the limitations and capabilities of the hardware used.

1.2 Additive Manufacturing Technologies

There are several AM technologies that have emerged over the years. All of the technologies work on the same basis of layered manufacturing, the differences among these technologies relate to the materials used, the equipment employed, the fabrication environment, and fabrication process or technique

1.2.1 Stereolithography

Charles (Chuck) Hull coined the term “stereolithography” in his patented method and apparatus that makes solid objects by printing thin layers of an ultraviolet curable material on top of one another. Figure 1-2 presents a concentrated beam of ultraviolet light is focused on the top surface and draws the object layer by layer through polymerization of monomer. The monomer in the ultraviolet curable material is polymerized into large molecules which are cross-linked to create a solid structure.

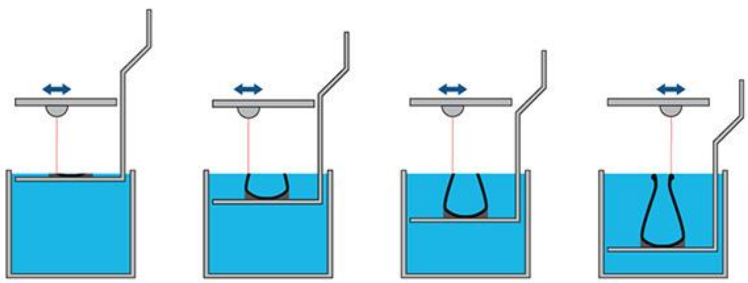


Figure 1-2 Stereolithograthy apparatus schematic [3]

The limitation of this technique is the strength of the object. The photopolymer-made prototype is brittle and may not be strong enough to withstand high stress testing. Also, the shrinkage of the material may deform the prototype. As the photopolymers are a thermoset material, they cannot be melted and reused. The accuracy achieved is about 0.1% of the overall dimension and deteriorates with larger sizes but no more than 0.5%. The layer thickness is between 0.004’ and 0.03” [1].

1.2.2 Selective Laser Sintering (SLS)

SLS uses a carbon dioxide laser to sinter successive layers of powder instead of liquid. SLS can be used with a variety of materials such as polymers, metals, ceramics and composites, including polycarbonate, PVC (polyvinyl chloride), ABS (acrylonitrile butadiene styrene), nylon, resin, polyester, polypropane, polyurethane and investment casting wax, which allows the resulting parts to be used for a variety of applications.

The average accuracy achieved ranges from +0.005” to +0.015” while the layer thickness is between 0.003” and 0.02”.

1.2.3 Fused Deposition Modelling (FDM)

In an FDM process as presented in Figure 1-3, a spool of thermoplastic filament feeds into a heated extrusion head. The movement of the FDM head is controlled by computer. AM machines that utilize this process typically feature a build platform and an extrusion head that is capable of processing build material typically supplied in form of a filament spool. Inside the extrusion head, drive wheels feed the filament into a liquefier where the filament is melted into liquid (at least 1°C above the melting temperature) usually by a resistive heater. The head traces an outline of each cross-section layer of the part. As the head moves horizontally (in X and Y plane) the thermoplastic material is extruded out of a nozzle by an extruder. The material solidifies in 1/10 second as it is directed on to the workplace. After one layer is completed, the extrusion head moves up a programmed distance (the layer height) in the Z-direction for building the next layer. Each layer is bonded to the previous layer through thermal heating.

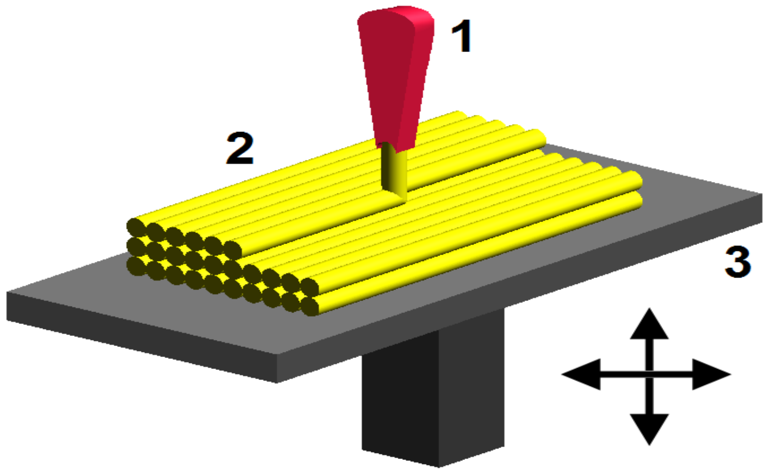


Figure 1-3 Fused deposition modelling [4]

1 – nozzle ejecting molten material, 2 – deposited material (modeled part), 3 – controlled movable table

The FDM technology accommodates a variety of modelling materials and colors for model building. Available materials are wax-filled plastic adhesive, proprietary nylon, investment casting wax etc.. There is minimum material wastage in the method. One advantage of this method is that it does not need any post-curing. In many cases, the FDM process does not need support to produce part since depending on the geometry of the model, the FDM extrusion head could form a precision horizontal support in mid-air as it solidifies. For overhanging parts, a support may still be required to limit part distortion.

1.2.4 Laminated object manufacturing (LOM):

Laminated object manufacturing (LOM) is a rapid prototyping system developed by Helisys Inc. The LOM processes produce parts from bonded paper plastic, metal or composite sheet stock. LOM machines bond a layer of sheet material to a stack of previously formed laminations, and then a laser beam follows the contour of part of a cross-section generated by CAD to cut it to the required shape. The layers can be glued or welded together.

1.2.5 Ballistic Particle Manufacturing (BPM)

The ballistic particle *manufacturing technique, devel*oped by Perception Systems uses a piezo-driven inkjet mechanism to shoot droplets of melted materials, which cold-weld together, onto a previously deposited layer. A layer is created by moving the droplet nozzle in x and y directions. After a layer is formed, the base plate lowers a specified distance and a new layer is created on the top of the previous one [1].

1.2.6 Three-dimensional printing (3DP):

With the 3D Printing process, a 3D model is sliced into 2D cross-section layers in computer. Figure 1-4 explains the process of 3DP where a layer of powder is spread on the top of the piston, the powder bed, in a cylinder, and then an inkjet printing head projects droplets of binder material onto the powder at the place where the solidification is required according to the information from the computer model. After one layer is completed, the piston drops a predefined distance and a new layer of powder is spread out and selectively glued. When the whole part is completed, heat treatment is required to enhance the bonding of the glued powder, and then the non-bonded powder is removed [1].

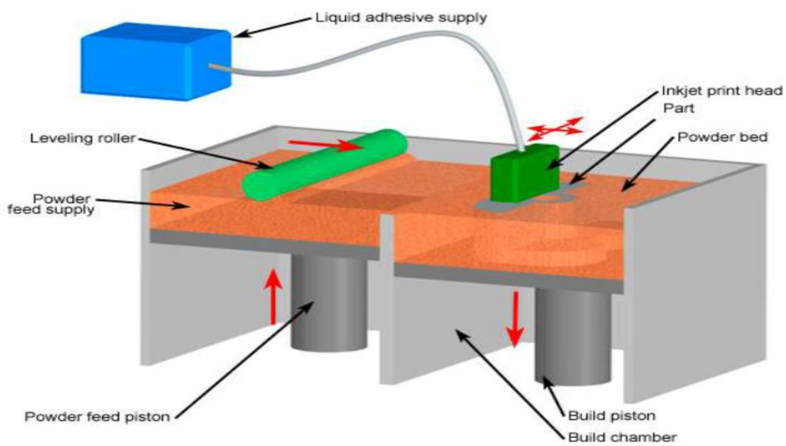


Figure 1-4 Working principle of Three-Dimensional Printing [5]

The 3D Printing process can use aluminum-oxide and alumina-silica ceramic powders. The binder material is amorphous or colloidal silicon carbide. With the 3D Printing technique, the design of support structure for the part is not needed, since the non-bonded powder of each layer remains to form a natural support during the layering process [1].

1.3 The RepRap Project

Dr. Adrian Bowyer, at the University of Bath, United Kingdom founded the RepRap project in 2005 as an initiative to develop a 3D printer that can print most of its own components. RepRap (REPlicating RAPid prototype) primarily uses an Fused Filament Fabrication (also known as Fused Deposition Modeling) additive manufacturing technique to lay down material in layers to create objects from thermopolymers such as polylactic acid (PLA) and acrylonitrile butadiene styrene (ABS). Since 2005 the RepRap project has resulted in the development of several prototypers. Apart from structural components holding all components together, the RepRap prototyper consists of an electronic board, extrusion head and stepper motors that drive the extrusion head [6]. The first RepRap, Darwin is seen in Figure 1-5.

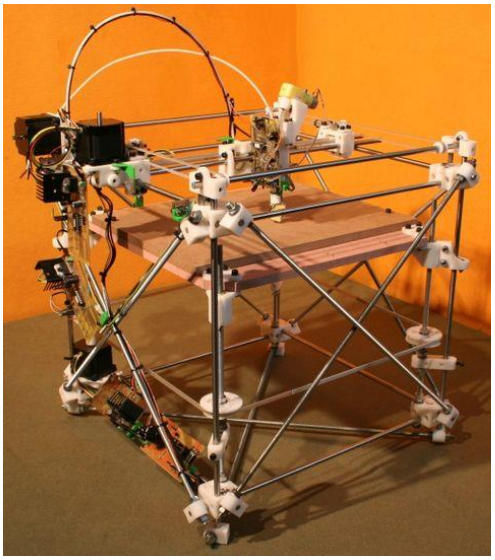


Figure 1-5 RepRap version 1.0 (Darwin)

RepRap Project is constantly being developed by developers all over the world due to its open-source nature. In the current state of development, most prototypers feature one single extruder with some going up to two extruders. Also the controlling software that drives the machines are capable only to extrude alternately but not simultaneously.

Chapter-1 explored the concept of Additive Manufacturing and summarizes some of the various additive manufacturing techniques. Finally the RepRap Project was introduced as it forms the basis of the open source nature of the platform discussed in Chapter 2.

Chapter 2

Custom Multi-Modality 3D BioPrinter

Layer-by-layer additive manufacturing provides the tools and ability to create prototypes from CAD models quickly, at lower costs and efficiently. This is further helped by open projects such as the RepRap project that has made the technology widely available to the general public at lower cost compared to industrial strength equipment.

Layer-by-layer manufacturing processes have become more popular commercially and for prototyping, those processes are restricted in several ways. The current common AM equipment (based on FDM or FFF) and technologies are capable of processing a single material or maximum up to two similar materials on 3D printers (Makerbot Replicator 2X), but they are restricted to a single fabrication (extrusion) process. Some additive manufacturing machines can use more than two materials however they are restricted to using the same material processing technology . The Stratasys Objet Connex 3D Printer series, which can print with as many as fourteen materials employing the same polyjet printing process or technology for all the materials. Yet an area where most of these machines lack is the ability to create functional structures with customized properties and heterogeneous materials and processes.

Many research groups have chosen to develop control software that has the ability to specify custom toolpaths that have the ability to deposit desired material at appropriate locations. The deposition of all materials takes place automatically after the print command is issued without any user intervention provided the positional data is valid throughout the manufacturing [7, 8, 9].

Loss of positional calibration in machines using multiple freeform fabrication techniques is a major shortcoming in additive manufacturing especially in the field of biomedical research where multiple processes maybe advantageously employed to manufacture constructs with custom properties.

The Custom Multi-Modality 3D Bioprinter (CMMB), presented in Figure 2-1 developed at the MARS lab that combines the Fused Filament Fabrication (FFF), Photo Polymerization, Viscous Extrusion (VE), and Inkjet printing technologies onto a single additive manufacturing platform. The construction and features of the CMMB are discussed by Ravi et al. [10].

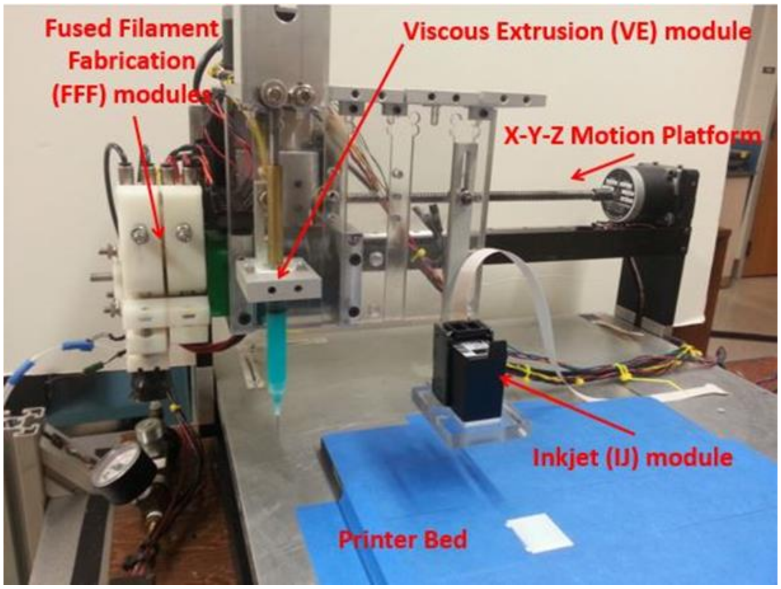


Figure 2-1 Custom Multi-Modality 3D Bioprinter

The CMMB was developed by transforming an existing machining microrouter (MicroKinetics) into an additive manufacturing platform. A RUMBA process control card provides the interface between the driving software and hardware [11]. The board sends motion commands to the stepper motors, sends signals to the heaters, receives signals from the thermocouples and sends digital output signals as needed. The open source software Repetier-Host and Slic3r provide the interface with and control the printer for fabrication purposes. The resolution of the motion platform is 0.003mm in the X, Y and Z directions and has the maximum feed rate of 10mm/sec

2.1 The Concept of Modularity

Modularity and full control over each process is one of the primary motivations of this research. The concept of modularity as applied here refers to distinct manufacturing heads (of the same/different processing technologies), i.e. a manufacturing head which can be independently commanded to controllably dispense corresponding material at the appropriate location [10]. Right from the initial design stage, the concept was to develop a modular expandable additive manufacturing fabrication environment and provide easy access to the hardware. Although the current (Nov 2015) setup of the CMMB includes two FFF and a single PP/VE and IJ modules, it can be easily expanded to include up to six FFF modules and up to a combined total of four PP/VE and IJ modules in any combination. The authors postulated that this capability of easy expansion of hardware coupled with features present in the open source software (Slic3r and Repetier-Host) will allow mixing and matching of different materials during fabrication and advantageously employ diverse material properties during fabrication to meet customized requirements and functionality.

2.2 Hardware Features

2.2.1 Fused Filament Fabrication Module (FFF)

An FFF module is used to create 3D structures by extruding a stream of heated or molten thermoplastic filament such as ABS or PLA. The raw filament, usually 1.75mm or 3mm in diameter, is pushed into a heated block where is heated at temperatures above 180°C (PLA) or 230°C (ABS) and pushed out through a nozzle. The nozzle has an exit diameter typically ranging between 0.1mm and 0.5mm. The extruded material is in the form of a cylinder that when deposited takes an elliptical shape due to being pressed down by the hot nozzle. The extruded material strands are either laid side-by-side or on top of another and immediately hardens on cooling while providing structural integrity.

Numerous modifications have been made to the CMMB to adapt it for multi-heterogeneous material and multi-process fabrication. Usually on RepRap machines, the extruder head is a monolithic unit and adjustments to nozzle to bed distance (along Z-axis) are either performed by lowering the bed or by raising the complete head. This setup was not suitable for CMMB since CMMB is a modular fabricator, each module on the printer has the capability to be individually calibrated and controlled.

Another modification that has been made on the CMMB is an actuation mechanism for the engagement and disengagement that allows for individual control of multiple FFF modules. RepRap machines that feature multiple extruders require that each extruder has its own stepper driver. This limits the number of extruders that can be used using a commonly available RepRap board. On CMMB that uses the RUMBA process control card, all the FFF modules are driven on a single stepper motor. Each stepper is engaged or disengaged as required using specific tool-change commands that engage the actuation mechanism to engage individual or multiple FFF modules.

2.2.2 Inkjet Module (IJ)

Thermal inkjet printing modules have been used in several research studies to controllably dispense cells [12, 13, 14] and conductive ink [15, 16].Due to the established track record of previous studies, the IJ module was incorporated on the CMMB in order combine its features with the FFF and PP/VE modules. The inkjet module(s) can be independently operated and calibrated.

2.2.3 Viscous Extrusion (VE) / PhotoPolymerization (PP)

The Viscous Extrusion/ PhotoPolymerization (PP) module consists of a syringe with a nozzle and a plunger attached to a stepper motor, mounted on a slotted guide way. The slotted guide way allows for independent calibration of the nozzle to bed distance.

2.2.4 RUMBA Process Control Card

According to the RepRapWiki, RUMBA (Reprap Universal Mega Board with Allegro driver) is a feature rich all-in-one electronics solution for controlling RepRap type 3D printers and CNC machines [17] The card can control up to six stepper motors with three reserved for the motion platform. It has provisions to heat the FFF modules and to monitor the current temperature on each extruder using thermistors. The heaters can be controlled using a PID controller that keeps the temperature within the desired range to minimize the variations.

The RUMBA process control card has the capability to run a number of RepRap compatible firmware to control the various functions of the fabricator. The Marlin firmware is used for the CMMBThe firmware was substantially modified to enable independent multiple module functionality needed on the CMMB.

2.3 Operating Software

To fabricate or print any structure using the CMMB and similarly any 3D printer, first a 3D model of the part needs to be created in a 3D modelling software such as Dassault Systemes SolidWorks, Autodesk 3DS Max and PTC Creo etc. Other options include non-mechanical CAD systems such as TinkerCAD, Blender, Art of Illusion, OpenSCAD, FreeCAD and Google SketchUp which are freeware and can be used by anyone without any license requirements. In this research SolidWorks by Dassault Systemes was used to create 3D models and to demonstrate design level strategies for the design and fabrication of multi-material and multi-modality 3D parts and assemblies.

Once the 3D part or assembly is prepared, it is then sliced to create a G-Code. Slicing (a RepRap term) which refers to analyzing the 3D solid model geometry and features to create instructions for the fabricator. Some popular slicing software include, Skeinforge, Slic3r, Cura and KISSlicer. In this research, Slic3r is used for its simplicity and numerous advanced features.

Usually the slicing software is a part of a larger software suite, usually called the "Host Software" which implements the slicer and a G-Code sender. These host software suites form a bridge between the solid model and the printer firmware and bidirectionally communicate with the fabricator by sending instructions to the fabricator (motion and actuators) and receiving information from the sensors. These instructions are in the form of G-Code, an industry standard for controlling the motions of CNC machines and adapted for additive manufacturing machines. Some popular host software suites include Pronterface, RepSnapper, Repetier-Host, Repetier-Server and OctoPrint. In this research Repetier-Host and OctoPrint are used as host software suites.

The last piece of software is the firmware. A firmware is flashed directly to the control board which interprets the incoming instructions (G-Code) and performs functions such as heating and motion. Popular firmware include Sprinter, Marlin, Teacup and Repetier. CMMB uses Marlin due to vast support from the open-source community and its flexibility to perform functions as needed.

The following section discusses some of the most used features of above mentioned software in each category,

2.3.1 Dassault Systemes SolidWorks 2015

SolidWorks 2015 is a solid modeling computer-aided design (CAD) program that provides several advanced tools to create intricate geometries and assemblies, shown in Figure 2-2, which can then be exported to STL (STereoLithograpy) or AMF (Additive

Manufacturing Format) formats. Both STL and AMF are the most widely supported formats in the world of additive manufacturing and will be discussed in detail in Chapter 3.

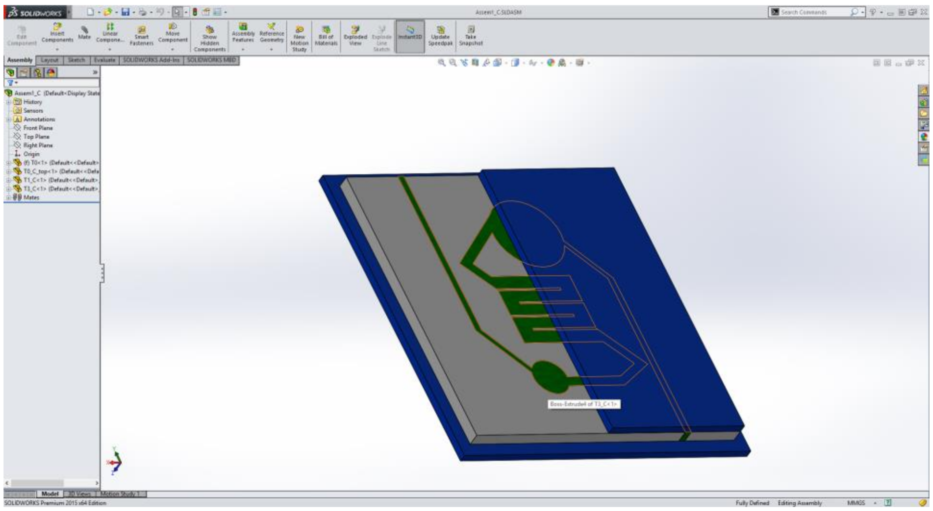


Figure 2-2 An assembly in Solidworks 2015

2.3.2 Slic3r

According to the RepRap wiki, Slic3r is the tool needed to convert a digital 3D model into printing instructions for a 3D printer [18]. Models are imported into the program from an STL/AMF generated file and are cut into horizontal slices (layers). Each slice is analyzed and a toolpath to recreate the slice is generated which calculates the movement of the tool-head in X, Y and Z direction and also how much manufacturing material should be extruded. Figure 2-3 shows the application interface of Slic3r.

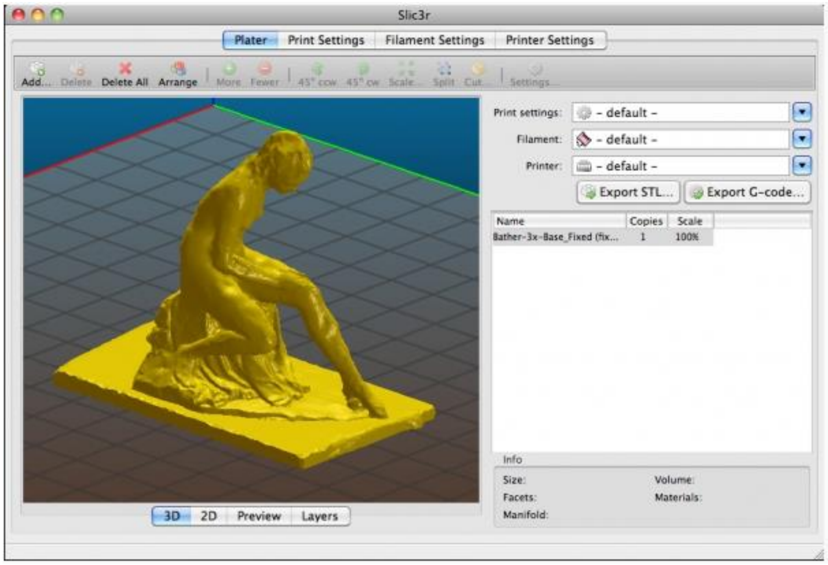


Figure 2-3 Slic3r interface [19]

Some of the major features of Slic3r include its support of multi-material (multiple extruders) object printing that allows creating structures with multiple modules. Multiple G-Code flavors are supported out of the box that allow Slic3r to create G-Code for a variety of firmware. The 3D preview shows the user the positioning and print parameters before printing, thus providing the opportunity for the user to perform modifications without actually printing a part.

Slic3r allows the user to control configurations of speed, acceleration, extrusion width and infill patterns (honeycomb, spirals, Hilbert curves etc). Customizable G-code macros and output filename with variable placeholders along with support for post-processing scripts makes Slic3r well suited for multi-material and multi-process additive manufacturing [18].

2.3.3 Repetier- Host:

Repetier-Host is used as the host software because of its full integration with Slic3r and an inbuilt G-Code viewer. It has the ability to connect with numerous types of machines and interact with them over a number of protocols.

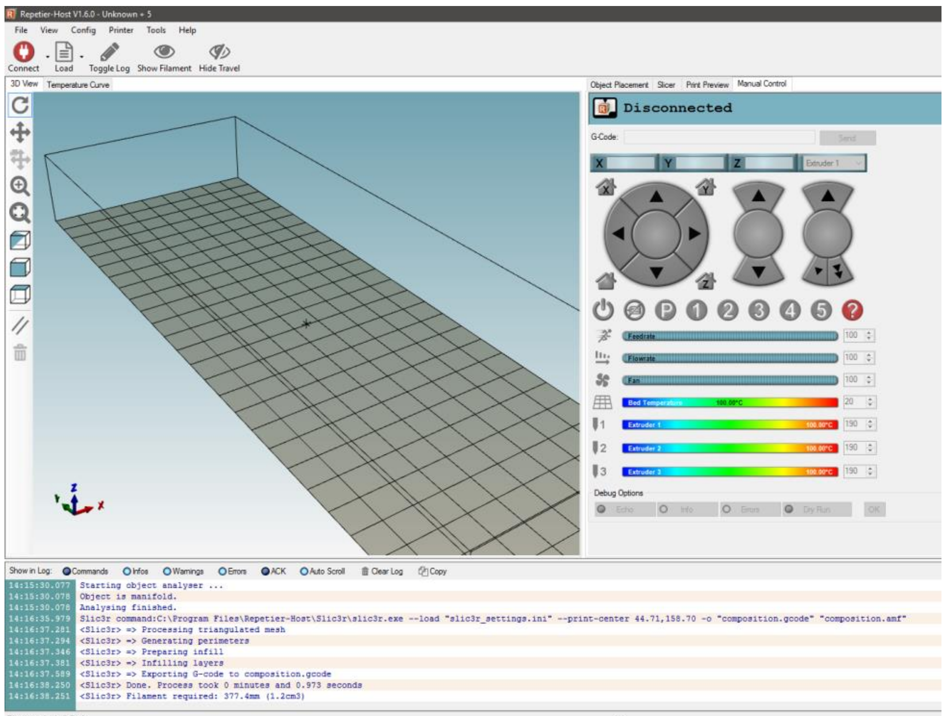


Figure 2-4 Repetier-Host interface

Repetier-Host provides for full control over the fabricator system where one could manufacture multiple objects and perform machine functions such as axis homing, heating individual extruders and manual tool switching. While performing these functions, Repetier-Host also communicates with the control board to get positional and thermal feedback. This ensures high and precise quality manufacturing.

2.3.4 Marlin Firmware

Apart from the tangible software discussed above to design 3D models and to control the machine, another important piece of software is the firmware that is flashed on the RUMBA process control card.

On the CMMB, Marlin was selected as the firmware. Marlin is a mashup between older firmware like Grbl and Sprinter. The firmware is needed to be flashed only once onto the process control card and it interprets the incoming instructions that are in the form of a G-Code. Once it has the instructions, the card then controls the motion and other functions of the fabricator accordingly. The firmware configuration is unique to each machine and all the properties such as dimensions, heating power and communication protocols are defined inside the firmware. Advanced functions such as retraction, levelling, PID control, and look ahead among others are stored in it.

Features such as Thermal Runaway Protection and Interrupt based temperature protection prevents overheating and damage to the modules. This safety feature is important in multi-modality setups where different modules are at different temperatures and a buffer failure can cause overheating.

A PID controller controls the heating of the modules to ensure stability and reduce thermal errors. Initially full current is supplied to the heater to ensure fast heating and when the temperature of the module is less than 10°C from the desired set value, PID controller takes over heating control. The PID controller gains are set by the M303 command.

. The Look-Ahead feature of the Marlin firmware requests the G-Code in advance and stores it in a buffer. This buffer is used to calculate the acceleration and deceleration of the modules to maximize speeds especially when approaching and leaving corners.

Marlin firmware includes a large dictionary of G-Code that can be used to perform a variety of functions. In its unmodified state, Marlin is suitable with most RepRap machines. With those machines only some changes relating to print volume and temperatures need to be performed. When using Marlin for CMMB, several other changes were made. Marlin, by default can only support up to four extruders [20] which limits CMMB's functionality and defeats the concept of modularity.

A modified version of the Marlin firmware that disabled the function to prevent cold extrusion was obtained from GitHub [21]. This was important for functions such as syringe extrusion where the material was not required to be heated. The firmware was further modified to allow seamless integration and control of multiple FFF modules with a single stepper motor. As a default, Marlin comes with 'Active extruder' turned on, which means that only one tool would be possible to use at a time. This meant that on the CMMB, if a second extruder were to be engaged, the single stepper motor being used to drive the multiple FFF modules would not function. This issue was addressed by removing the 'Active Extruder' concept from the firmware.

Table 2-1 Defined tool change commands for independent modules and their axis on the CMMB

Tool Change Command	Function	Extrusion Axis
T0	FFF – Extruder 1	E
T1	FFF – Extruder 1	E
T2	Inkjet Module	N/A
N/A	VE/PP – Module 1	I
N/A	VE/PP – Module 2	J

Other changes included redefining the tool change commands (T0, T1, T2, etc.) in the Marlin_main.cpp file in order to provide the ability to control individually or in multiples the FFF modules a single filament feeder motor. The T0/T1 tool change commands were set to engage the two corresponding FFF modules one at a time, whereas T2 was defined to activate the IJ module in the modified firmware. Various toolchange commands, with their respective extrusion axis and function have been discussed in Table 2-1

Chapter 2 introduces the Custom Multi-Modality 3D Bioprinter (CMMB) with its features and discusses the concept of modularity as the design philosophy of CMMB. Various software used are also discussed along with their important characteristics.

Chapter 3

Additive Manufacturing File Formats

Solid models created in any modelling software (Solidworks, PTC Creo etc.) need to be exported in mesh format for additive manufacturing. Meshing refers to the triangulation of a plane or volume. A triangulation of a planar object is a subdivision into triangles, and by extension the subdivision of higher-dimension geometric objects into 'simplices'. Triangulation of a three-dimensional volume would involve subdividing it into packed 'tetrahedra' ("pyramids" of various sizes) [22] [23]. This is a basic requirement in the field of manufacturing and prototyping as the processing software needs to know and analyze the sub-structure of the structure to successfully create a toolpath for the deposition head. Most common additive manufacturing formats include STL, AMF, OBJ and 3MF. AMF and 3MF are the only formats to inherently support multiple material and multiple process definitions.

3.1 STL (STereoLithography) File format

STL file format is supported by many CAD software packages and is widely used for rapid prototyping, 3D printing and computer-aided manufacturing. The file format was created by 3D Systems for the Stereolithography CAD software [24] [25] [26]. Stereolithography and additive manufacturing machines require a series of 2D closed contours that are filled with solidified material as the layers are fused together. When a model is exported to STL format, the complete model is split along the X-Y plane at different Z values (height) where the thickness of the slice is the difference between the next and last Z values. The thickness slice is the layer height and determines the build resolution along the Z-axis.

The STL format has the capability to define polyhedrons, but in practice it only defines triangles. For curves, STL divides the contour into a triangular mesh which is an approximation of the original CAD model. The precision can be increased by increasing the resolution of the mesh. This is accomplished by increasing the number of mesh elements (triangles). CAD software has the ability to adjust the mesh fineness. However, it is important to note that even though an infinite number of triangles could be defined, there is a limitation on the size of the triangles due to hardware motion limitations and process hardware.

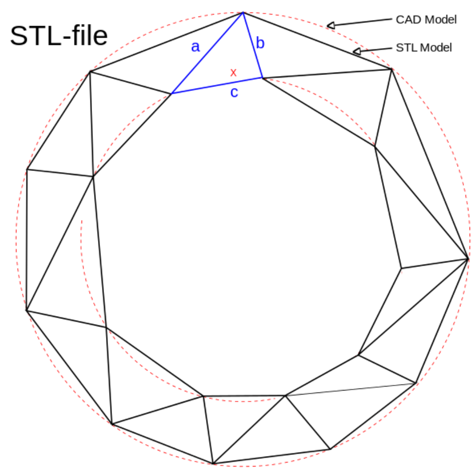


Figure 3-1 The differences between CAD and STL models [27]

Figure 3-1 presents the differences between CAD and STL models. When the model is exported to STL format, the donut surface is divided into a set of triangles with a restriction that each triangle side is entirely shared by maximum two adjacent triangles in a process called ‘triangulation’ [28]. Theoretically, every surface could be triangulated, but it might require an infinite number of triangles. The number of triangles in triangulation is limited by the desired or defined mesh resolution, seen in Figure 3-2.

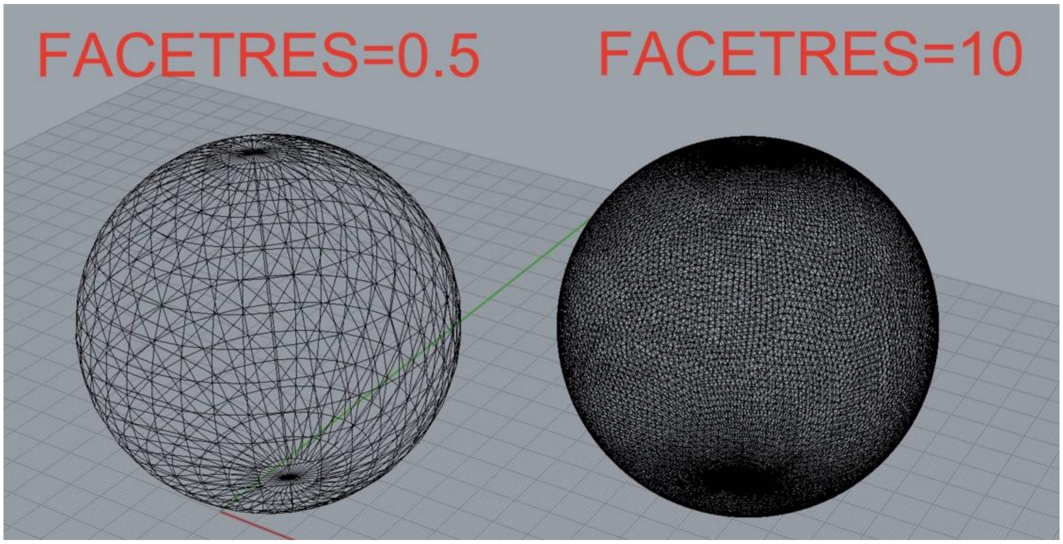


Figure 3-2 Comparison of two mesh resolutions [29]

Inside the STL file, the object represented must be located in the all-positive octant. In other words, all vertex coordinates must be positive-definite (nonnegative and nonzero) numbers. The coordinates have arbitrary units as the STL file does not contain any scale information [28]. Apart from scale information, STL files do not contain any object, texture or color information. A mesh exported to STL lacks information to define individual objects in multi-object assembly. This is a major limitation for manufacturing using multiple materials and processes.

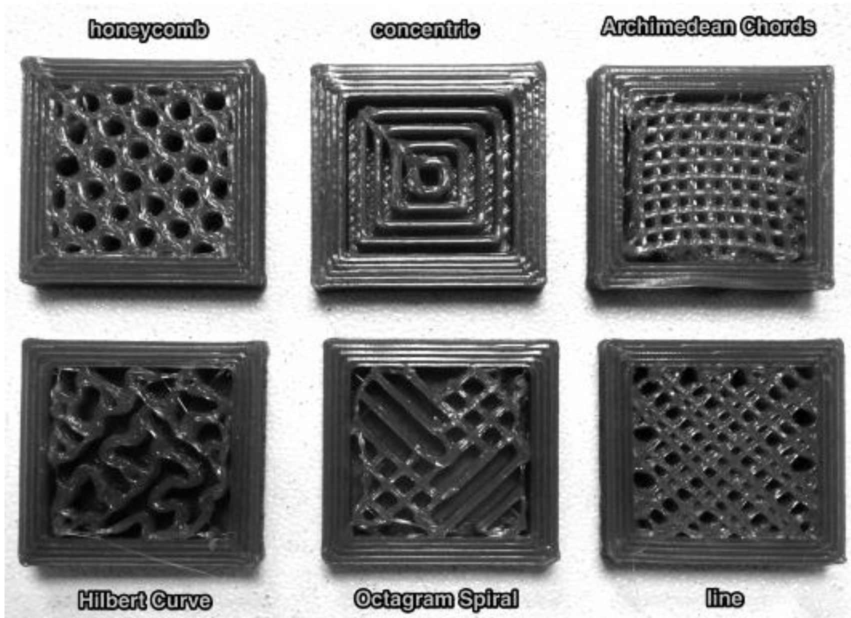


Figure 3-3 Different infill types in Slic3r

Figure 3-3 shows the cross-section of a cube exported as STL and sliced using Slic3r with different infill settings, which shows that the printed object is homogenous in nature and Slic3r provides no control over regional properties. In other words, the settings are applied globally to the complete model when sliced using Slic3r.

3.2 3MF File Format

An industry consortium setup to define a new 3D printing format that will allow CAD software tools (SolidWorks, AutoCAD, Rhino, CATIA, etc.) to send the full 3D model to 3D printers from different manufacturers and other platforms. The 3MF Consortium, launched in 2015, aims at quickly releasing and maintaining a specification that allows companies to focus on innovation rather than on basic interoperability issues [30].

The 3MF-C uses the ".3MF" file format developed independently by Microsoft and donated to the consortium. The ".3MF" is more of a container format developed over the ".

AMF” to simplify the print process. The Consortium’s website (http://www.3mf.io) [30] describes .3MF as a 3D model and the contained format called the 3MF Document. The format requirements are an extension of the packaging requirements described in the Open Packaging Conventions specification [31]. That specification describes packaging and physical format conventions for the use of XML, Unicode, ZIP, and other technologies to organize the content and resources that make up any model, and are an integral part of the 3MF specification.

The 3MF format is relatively new and is yet to be implemented in many of the popular design and slicing software systems. At the time of writing of this thesis, it is not known if post-processing could be performed on 3MF files as the print ticket is packaged within the 3MF file.

3.3 AMF File Format

The ASTM Committee F42 on Additive Manufacturing of the American Society for Testing and Materials, in 2011 developed the AMF format for technology independence, simplicity, performance and both future and backward-compatibility. The International Standards Organization (ISO) approved the format in 2013. AMF is the successor to STL and unlike its predecessor has native support for color, materials, lattices and constellations [32].

As described in the AMF specification, AMF is an XML based open format, where the file begins with the XML declaration. The file is primarily divided into five top level elements [32]:

1. Object: The volume or volumes of the material are defined in the object element, and each is associated with Material ID.
2. Material: The material element defines one or more materials.

3. Texture: The texture element defines the color or textures associated with the Material ID.
4. Constellation: Multiple objects are defined in a constellation. A constellation may have other constellations grouped into a relative pattern for manufacturing.
5. Metadata: The metadata specifies additional information about the objects or any of the above elements contained in the AMF file.

The AMF file format has the following characteristics:

1. The XML based open format allows for easy reading, writing and processing of AMF file with as much information as possible to describe an object and its properties.
2. Incorporates triangular meshes to describe surfaces which allows all the surfaces to be defined. Older STL format only allows single vertices to be defined.
3. AMF can define curved surfaces for better curve accuracy.
4. Objects can be defined in CAD programs allowing the user to differentiate each section of the model and assign required properties and materials to each section.

The above-mentioned features make AMF the most suitable format for defining methodologies and post-processing scripts. The AMF format, although slowly adapted by industry, has already been implemented in SolidWorks. SolidWorks 2015 can export AMF with colors and materials but Slic3r and Repetier-Host lack the ability to read those elements.

File formats such as STL, AMF and 3MF were discussed with their features and limitations. AMF was found to be best suited for this research as it can define multi-object and multi-material definitions. The objects within the AMF can be assigned properties and material and this has been discussed in Chapter 4.

Chapter 4

Multi Material and Module Additive Manufacturing

The CMMB developed at the MARS lab at The University of Texas at Arlington [10, 14] with its multiple modules and modular capabilities offers a perfect test bed to develop and test multi-material and multi-modality methodologies and strategies to create structures using additive manufacturing having heterogeneous geometries and materials. Functional components with heterogeneous properties have numerous applications in various engineering and biomedical fields.

4.1 Applications of Multi-Modality Additive Manufacturing

One of the motivations for this research, in terms of pragmatic application, is the creation of scaffolds for cell proliferation and growth found in the biomedical domain. Scaffolds play an important role in the regeneration of artificial tissues or organs. A scaffold is a porous structure with a micro-scale inner architecture in the range of several hundreds of micrometers. Figure 4-1 shows an image of biphasic silk and ceramic scaffold.

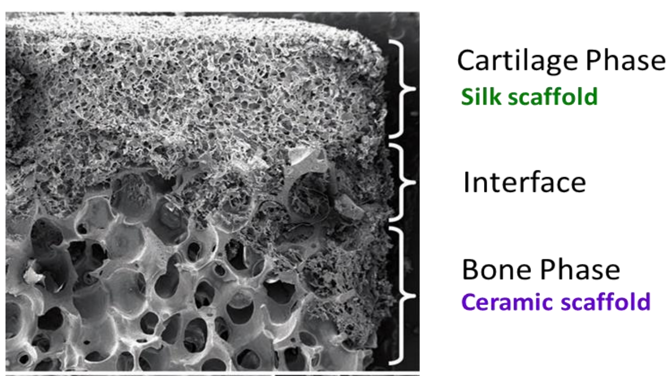


Figure 4-1 Biphasic silk and ceramic scaffold is designed to mimic properties of cartilage and bone for grafting into damaged joint [33]

A lot of difficulties are encountered when performing medical interventions or replacing organs according to “Unit cell-based computer-aided manufacturing system for tissue engineering” [34]. In such cases, artificial organ and tissue development using tissue engineering has gained attention as a viable solution. The three key components of tissue engineering are cells, cell signal, and scaffold. Cells are harvested from the human body and seeded into three-dimensional (3D) porous scaffolds. The cells then proliferate and differentiate into the target tissue in response to specific signals within the scaffold [35]. The seeded cells depend on the scaffold for enough physical strength to protect the immature tissue from external loads. Many studies have demonstrated a strong relationship between the inner architecture of a scaffold and tissue regeneration [36, 37, 38, 39] and have further focused on developing computer-aided technology [40, 41, 42] that allows for automated scaffold design which can have desired external and internal structures.

Current additive manufacturing technologies possess hardware to create homogeneous material based scaffolds, but lack the software to create the right toolpath. No open-source platforms that are popularly developed and used have features to define heterogeneous porous structures created with multiple materials. Some commercial equipment offer the possibility to create biocompatible constructs but are mostly closed source and limit modifications to toolpaths and provided materials.

The CMMB allows researchers to add or remove modules as needed while at the same time control each module independently. The software (including slicing and host) used is the same software used for other open-source equipment. Toolpath creation (slicing) is performed using Slic3r (http://Slic3r.org/) while Repetier-Host (http://www.repetier.com/) and OctoPrint (http://octoprint.org/) are the chosen host software. Both Repetier-Host and OctoPrint interact with CMMB and allow one to visualize

G-Code in advance and to understand the extrusion process and resolve any issues that might arise during the print. Slic3r helps in creating the toolpath but is primarily restricted to creating structures with homogenous geometries using the FDM/FFF extrusion. For other modules, the final G-Code need to post processed, which will be covered in a later chapter.

4.2 General Manufacturing Approach and Cautionary Steps

When manufacturing any structure using Additive Manufacturing, a 3D model is created using a 3D modeling software. Any 3D modeling software (Dassault Systemes SolidWorks, CATIA or PTC Creo, etc.) could be used but in this research Dassault Systemes Solidworks is used since the later versions include support for exporting the final model to an AMF format, one of the additive manufacturing STL, OBJ, AMF and 3MF formats. All these formats store the model geometry as a mesh and are readily readable by most slicing software. However, in order to define multiple sections for different manufacturing processes, AMF or 3MF must be used. At the time of researching multi-process methodologies, AMF was preferred due to its open nature and for its wider acceptance over 3MF.

While preparing the AMF for multi-bodied solid structure, the following prerequisites have to be adhered to:

1. <u>Separate Solid Bodies</u>: The multi-bodied part must have each material represented by an individual part or an assembly.
2. The individual parts or assembly in the multi-body assembly must have distinct regions identified with <u>no overlapping geometries</u>.
3. The complete geometry or any of the sub-geometry must be a manifold. A manifold structure refers to a structure that is water-tight and does not have any free edges.

These prerequisites must be incorporated during the design stage to ensure that structure can be correctly exported to an AMF file and the tool path generated would not contain errors or unwanted artifacts. Having distinct sub-geometries allows one to assign extruders/processes as required.

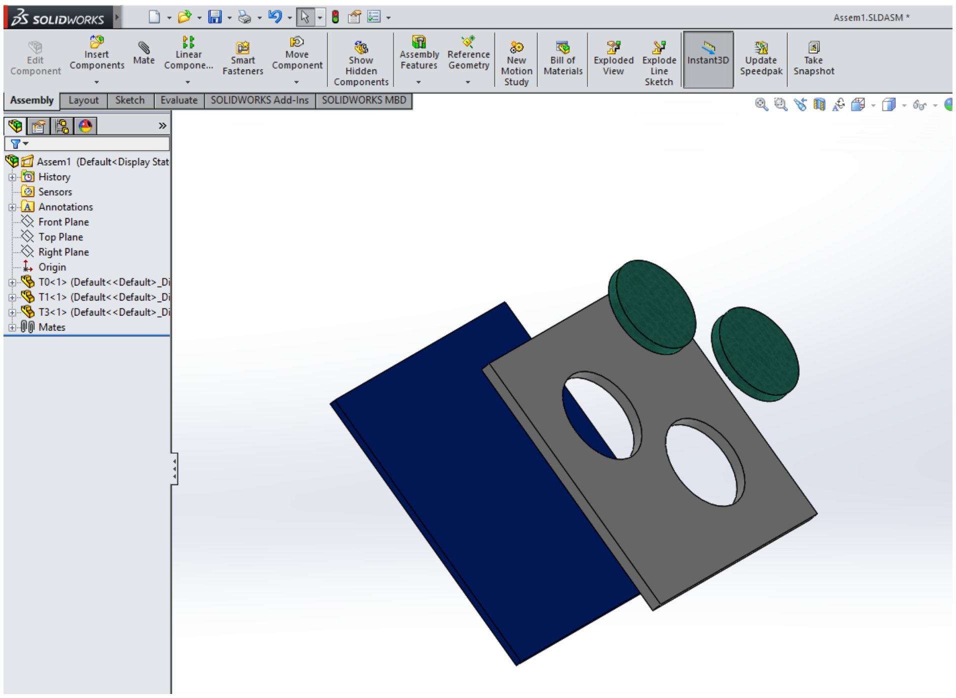


Figure 4-2 An exploded view of an assembly in Solidworks 2015

The following steps outline the generic process to design a 3D model for manufacturing and export it to AMF.

1. Open SolidWorks 2015 and start sketching on the “Front Plane”. This is important as “Front plane” corresponds to the X-Y plane of Repetier-Host. This ensures that the object created has the correct orientation when exported to STL/AMF.
2. Create a 2-D geometry from the origin. Once the sketch is completed for the first part, apply the features as needed and save the part.
3. Repeat step 2 for any additional part(s).

4. Now create an Assembly of all saved parts using mates. Once the assembly is fully defined, save it.
5. Export the AMF using File -> Save as -> Select Additive Manufacturing File (.AMF) as file type. Enter the file name and press ‘Save’. Note that the correct unit needs to be selected using the ‘Options’ button
6. The AMF exported in step 6 can now be imported in Repetier-Host.

Once the AMF is generated, it needs to be converted into a G-Code that contains the toolpath and extrusion parameters using a slicing program such as Slic3r. Slic3r is available both as an integral part of Repetier-Host and as a standalone application. The following steps give an outline on importing the AMF into Repetier-Host and then slicing it using Slic3r to get a toolpath.

1. Open Printer Settings -> Extruder. Increase the number of extruders to desired number and assign individual properties such as temperature and offsets along X and Y axes measured relative to T0.
2. Import the prepared AMF and combine individual parts into the same Object Group using the Settings button. If the imported file is STL then the file will be imported as a single object that does not contain information about individual parts.
3. Assign extruders as desired to individual objects within the Object Group as shown in Figure 4-3. If the imported file is STL then extruders cannot be assigned.

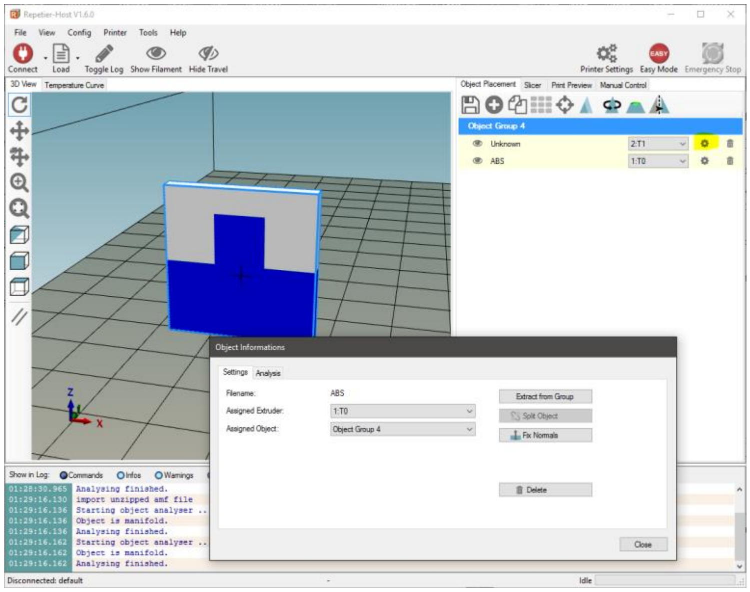


Figure 4-3 Assigning extruders and grouping objects

4. Inside the ‘Slice’ tab, seen in Figure 4-4, choose Slic3r as the slicing software and choose the defined profiles for Print Settings, Filament Settings and Printer Settings. The profiles are machine specific and include parameters such as infill density and type, nozzle diameter, speed and custom G-Code among others. More information on different settings can be found in Appendix-A. Make sure that the extruder settings and offsets are defined inside the Slic3r profile as well.
5. Once the settings are chosen, press the ‘Slice with Slic3r’ button. This will process the model and create a toolpath ready to print.

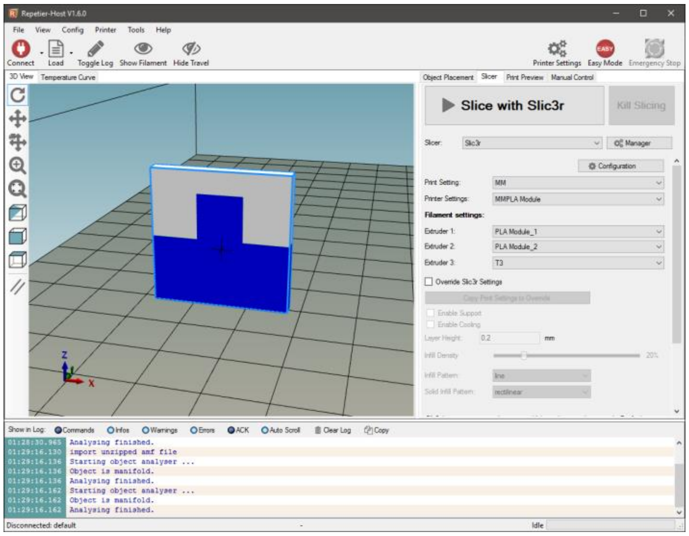


Figure 4-4 Slicer tab in Repetier-Host

6. In the preview tab, the G-Code can be visualized and any manual edits be made.
7. Press ‘Print’ after making modifications, if any.

Models exported in STL are not suitable for multi-process additive manufacturing as they do not contain information about materials and individual objects. STL based structures are primarily homogenous and without the ability to define multiple properties over the structure.

Exporting a model as an AMF results in a file that contains information about individual objects and materials. This information can be used to assign extruders which may be used to process different materials. For example, Extruder-1 (FFF) may have soft PLA, while Extruder-2 (FFF) may have hard PLA. Other combinations may include PLA and ABS or ABS and Nylon etc. Multiple processes can also be assigned to individual parts. For example, a structure can be created that is made up of FFF with two or more thermoplastics and Viscous Extrusion (VE) with photo-polymerization resin.

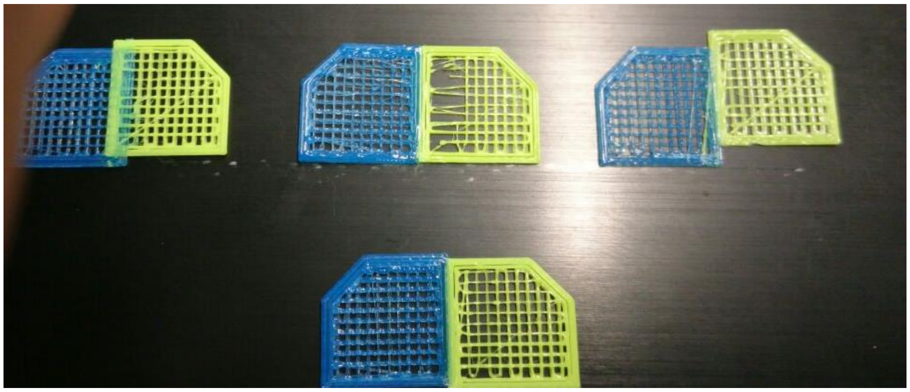


Figure 4-5 Printed multi-material structure using two STLs

Figure 4-5 shows a multi-material structure created using two STLs that are placed next to each other in Repetier-Host and sliced using Slic3r after assigning different FFF extruders. Positioning of the two individual STL required repeated calibration with respect to each other as the second extruder was at an offset from first extruder. Positioning of objects using trial and error required multiple prints to get the structure as close to the model, but this had zero percent repeatability. The placement and calibration become more difficult if different processes like VE and FFF are used in conjunction with one another, as high positional error would cause print failure.

AMF files on the other hand preserve information including multiple object position with respect to one another along with other properties such as material type and color. When modeling the structure in Solidworks, the assembly (.SLDASM) needs to be saved as a part file (.SLDPRT). Solidworks converts each part as sub-part from an assembly when it is saved as a part and preserves the displacement of individual objects from one another rather than the origin.

Printing an object using AMF file, allows us to print multi-process and material structures without having to adjust the relative position with one another as shown in Figure 4-6. The model is imported and printed on CMMB as described in Section 4.1. Complex and intricate structures can also be created by using AMF without performing repeated positional calibrations.

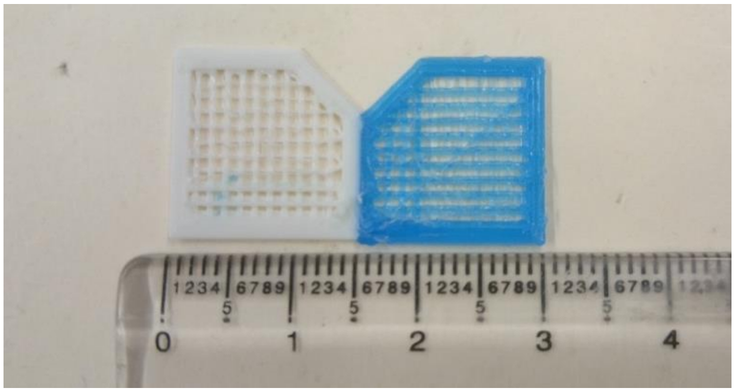


Figure 4-6 Structure created using AMF file format

One of the issues faced during print jobs was that some horizontal strands were not printed completely as shown in Figure 4-7. Such behavior was dominant immediately after switching extruders from T0 to T1 or vice-versa.

Figure 4-7 Missing horizontal strands

Further tests and observations were performed, after which it was determined that the missing filament strands was caused because the toolchange command (T#) was processed even before the previous extruder had finished extruding. The premature switching of the tool resulted in stop of flow of filament in the current tool, while the other tool would start extruding.

The Marlin firmware uses a ring buffer for its Look-Ahead feature. The Look-Ahead feature of the Marlin firmware requests the G-Code in advance and stores it in a buffer. This buffer is used to calculate the acceleration and deceleration of the modules to maximize speeds especially when approaching and leaving corners. With buffering enabled, G-Code commands ‘G0-G3’ (motion), ‘G28-G32’ (homing and calibration) and G90-92 (positioning) are buffered to minimize delay while a command is acknowledged and the next one is transmitted. Any other G-Code is not buffered. ‘T’ and ‘M’ G-Code commands bypass the buffer and are executed instantly resulting in an early toolchange switch.

In order to address this issue, an unbuffered G-Code command is used before the toolchange command. This causes the flow of commands to break as no acknowledgement is sent back to the host until the buffer is exhausted and the last command is executed. The host pauses and does not send any more commands until the last command is executed after which the toolchange command is executed.

The temperature command ‘M109’ was selected to be used before the toolchange command so that it can ensured that the extruder is at the right temperature before extruding. As ‘M109’ is an unbuffered ‘M’ G-Code command, it breaks the flow to the buffer and is only executed once the buffer is empty. This is followed by the toolchange G-Code command ‘T’ which activates and deactivates the tool at the right instant. The temperature command addition is suitable for FFF modules, which are equipped with heater and thermistor controls. For VE and other non-thermal based modules, the temperature command would fail as there was no feedback to the RUMBA process control board from those modules.

The temperature command was subsequently substituted by a pause G-Code command that paused the print for few milliseconds and would also break the flow control. The command used was ‘G4’. The pause command is applicable to all modules including non-thermal modules (e.g. VE).

A post-processing script was created in-house and experimentally verified using awk. awk is an interpreted programming language consisting of a set of actions to be taken for sets of textual data, executed either directly on input files or used as part of a pipeline, for purposes of extracting or transforming text. The basic function of awk is to search files for lines (or other units of text) that contain certain patterns. When a line matches one of the patterns, awk performs specified actions on that line. awk continues to process input lines in this way until it reaches the end of the input files. Programs written in awk are data-driven, in which the data is pre-described along with actions that are to be performed on the data [43]. In the case of post-processing G-Code, the G-Code are the data and post-processing are the actions. These actions are series of rules that look for the defined pattern and performs one or one set of actions upon finding the pattern.

On the CMMB, the post-processing was created using awk to add the pause G-Code command so that the print would pause for 500ms (maybe varied as needed) before executing the toolchange command. Figure 4-8 shows the pause command before the toolchange command.

```
G1 X127.705 Y151.279 E2.30417
G1 F480
G1 X127.703 Y152.612 E-2.44583
G1 E-2.69583 F1800.00000
G92 E0
G4 P500          ;Pause to stop buffer
T1
G92 E0
G1 X127.605 Y183.675 F600.000
G1 E6.00000 F1800.00000
G1 X113.605 Y183.675 E6.41568 F600.000
```

Figure 4-8 Pause command to flush the buffer before toolchange command

Table 4-1 Modified tool change definitions with their function and extrusion axis

Tool Change Command	Function	Extrusion Axis
T0	FFF – Extruder 1	E
T1	FFF – Extruder 2	E
T2	FFF – Extruder 3	E
T3	VE/PP – Module 1	I
T4	VE/PP – Module 2	J
T5	FFF – Extruder 1 and 2	E
T6	FFF – Extruder 2 and 3	E
T7	FFF – Extruder 1 and 3	E
T8	FFF – Extruder 1, 2 and 3	E
T9	Inkjet Module	N/A

The CMMB consists of inkjet and Viscous Extrusion (VE) modules in addition to FFF modules. For independent control of all modules, some global definitions were defined for correct module addressing. These global definitions ensure that the tools are switched and activated with their respective parameters which include temperature, extrusion axis, dump area etc. Toolchange commands (T0, T1, etc.) were selected for global definition. The Marlin firmware for the RUMBA board was modified with these global definitions as shown in Table 4-1.

By default, the toolchange commands enable the required extruder, while disabling any other extruder. As the concept of active extruder has been disabled on the CMMB, the toolchange commands were configured to activate and deactivate the extruder actuators. Specific control pins were defined in 'pins.h' while toolchange routines and functions were added in 'Marlin_main.cpp'. 'I' and 'J' extrusion axis were added by Ravi et al. to support Viscous Extrusion [14].

Commands 'T5 – T8' allow for the concurrent control of the FFF modules in any combination of two or all three thus enabling the manufacture of the same structure in multiples by harnessing parallel manufacturing illustrated in Figure 4-9. The CMMB can support multiple modules that can be used to run simultaneously to print multiple copies of the same object.

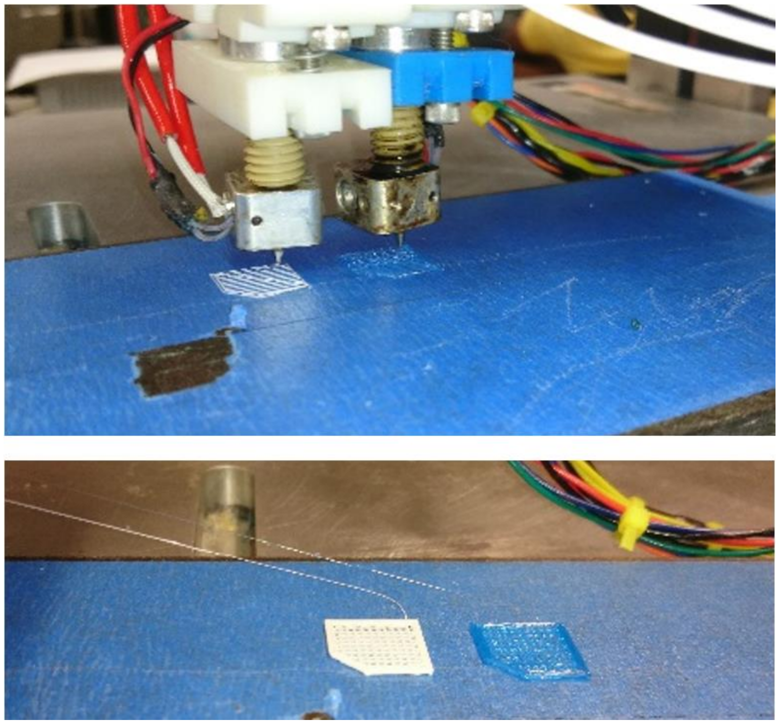


Figure 4-9 Parallel manufacturing using two FFF modules

On the CMMB, the use of Slic3r's generated G-Code, was limited to only the FFF modules. This is because the Slic3r generated G-Code consists of "E" as the only extrusion axis. On the other hand, the CMMB is a multi-modality printer that has other extrusion axes for driving the Viscous Extrusion modules. The viscous extrusion module uses additional extrusion axis including, "I" and "J". This is a result of the modification to the Marlin firmware with which seamless switching between extruders is possible.

The awk based post processing script was executed through a batch file that pipes the G-code through the AWK script and searches the G-code for tool change "T" commands and replaces subsequent "E" commands with desired extrusion axis name as was defined in the modified firmware. The final G-Code then can be used to manufacture structures on the CMMB using all modules as required. The working of the awk script is described starting with Figure 4-10. The script is called through a batch script where the relative path of the script, the input file and the output file are defined with any runtime options. Assuming that the following is the input G-code.

```
42  G1 X99.780 Y71.258 E1.61050 F1800.000
43  G1 X103.026 Y70.408 E0.10385
44  G1 X143.026 Y70.408 E1.23765
45  G1 X144.028 Y70.484 E0.03108
46  G1 X146.265 Y71.228 E0.07293 F1800.000
```

Figure 4-10 Sample Input G-Code

At the beginning of the AWK script, a field separator called ‘FS’ is defined in the BEGIN section of the awk script as FS=’ E’. This defines a point where each line of the G-Code is split into separate fields. AWK uses ‘ E’ as the field separator and splits each line as shown in Table 4-2.

Table 4-2 Field separation of G-Code using awk

Line	**$1**	**FS**	**$2**
42	G1 X99.780 Y71.258	E	1.61050 F1800.000
43	G1 X103.026 Y70.408	E	0.10385
44	G1 X143.026 Y70.408	E	1.23765
45	G1 X144.028 Y70.484	E	0.03108
46	G1 X146.265 Y71.228	E	0.07293 F1800.000

This separation of fields splits the G-Code line in two sections that are stored in ‘$1’ and ‘$2’. This is performed on all lines that contain the field separator (FS), which is ‘E’ in this case. The separated fields make it easier to put back information after the actions are performed on the G-Code.

As the awk script processes the G-Code, it searches for the toolchange command, “T#”, where ‘#’ is the assigned tool number. The tool numbers are assigned in the Marlin firmware. When a toolchange command is found in the G-Code, the script assigns the defined extrusion axis to a variable ‘EX’. The extrusion axis and individual toolchange commands of various modules are listed in Table 4-2. For the CMMB, T0, T1 and T2 are assigned to FFF modules, which use the extrusion axis ‘E’. The VE modules use T3 and T4 and have extrusion axis ‘I’ and ‘J’ respectively. Toolchange commands T0 to T2 and T5 to T8 activate the respective actuators while deactivating others while any other tool change command deactivates all of them.

The value of variable ‘EX’ does not change until the next toolchange command after which it takes the value of the next extrusion axis. Any G-Code command after the toolchange command that contains ‘E’ is processed and an output command prints out each line in the format defined in such a way that it includes the first part ($1), then the new extrusion axis, followed by the second part ($2). This is demonstrated in Figure 4-11, in which the extrusion axis E is changed to desired extrusion axis. In this case, the preceding toolchange command is T3 and the extrusion axis is changed to ‘I’

```
G1 X99.780 Y71.258 I1.61050 F1800.000
G1 X103.026 Y70.408 I0.10385
G1 X143.026 Y70.408 I1.23765
G1 X144.028 Y70.484 I0.03108
G1 X146.265 Y71.228 I0.07293 F1800.000
```

Figure 4-11 Output G-Code showing the new extrusion axis ‘I’

Slic3r by default gives G-Code that are suitable for FFF manufacturing. In VE extrusion, the syringe and nozzle have a large difference in size (diameter). Moreover, effects of capillary action and viscosity make it difficult to calculate the correct feed-rate for these modules. The awk based post-processing script was further developed to incorporate a multiplier “M”, to account for the correct feed rate. The multiplier can be applied independently to each module as needed and its value is dependent on the toolchange command.

Before the AMF can be sliced, all the modules must be given an offset from ‘T0’ which is considered to be at (0,0). The ability to use multiple modules simultaneously was used to calibrate the CMMB printer. An STL, having a ‘L-shaped’ model was sliced after resetting the offset of all modules to 0 mm. For calibration, the infill was set to 0% in Slic3r and the number of perimeters limited to one. The L-shaped model has a cross-section width equal to the nozzle diameter of 0.4 mm. The structure was parallel printed by enabling all modules and offsets measured along the X and Y axes. This structure is used for calibration purposes using image software tools.

Figure 4-12 ImageJ is used for calibration

Figure 4-12 shows ImageJ (http://imagej.nih.gov/ij) that was used to accurately measure the offsets between modules based on the fabricated calibration structure.

ImageJ is an image manipulation software available in the public domain and distributed by the National Institutes of Health, USA. On the CMMB it is recommended to perform the offset measurement each time the Z-height was calibrated. The whole calibration routine ensured that the manufactured structure did not have gaps. Calibration of the CMMB is an important step before printing an object. Without calibration, the pieces have irregular geometry with wrongly placed material as shown in Figure 4-13.

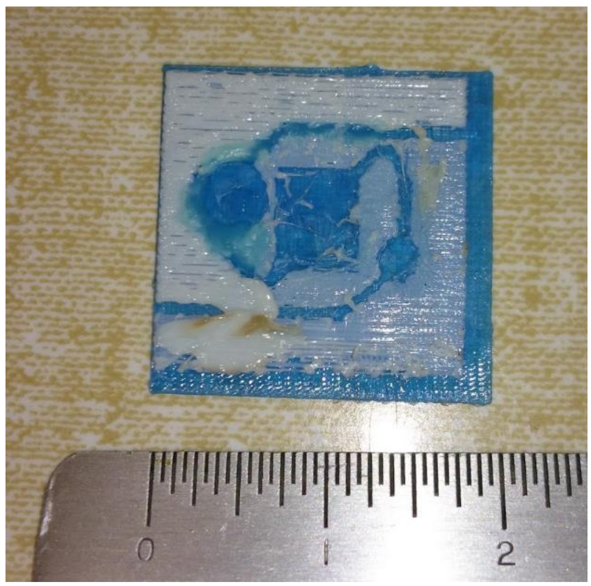


Figure 4-13 Failed builds because of non-calibrated CMMB

Several complex components were constructed with the CMMB using multiple modules and materials. On the CMMB, components used combinations of soft and hard PLA that introduced heterogeneous properties to the complete structure.

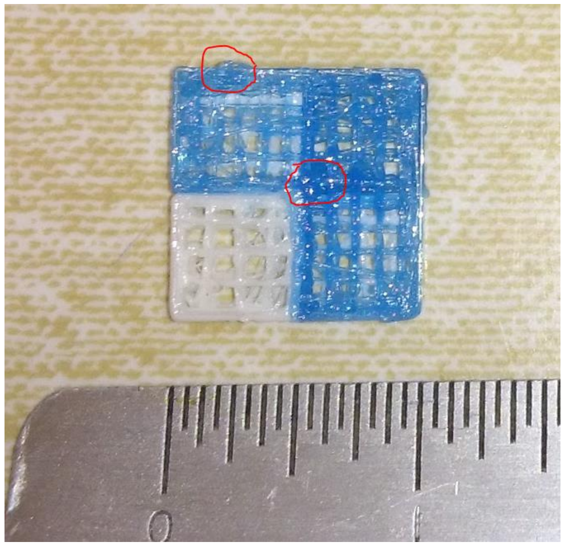


Figure 4-14 Structure with 1/4th region soft PLA and remaining region hard PLA (40% infill). Notice the ooze in the middle and edges (circled in red)

During module switching, oozing was found to be the second major problem as it deposited excess material over the top surface. When the next layer is printing, the extruder head would drag the oozed plastic over the piece, depositing it over other regions. In Figure 4-14 and Figure 4-15, the toolchange occurred within the same layer. The blue extruder dragged pieces of white and deposited it around the edges. The ooze in the middle is due to blue PLA being deposited over previous ooze. Usually the ooze occurs at start-points and end-points within the layer, and at points where modules are changed. When two materials are stacked over one another oozing is mostly not visible on the top surface but causes dimensional deformation on the sides, especially at the interface.

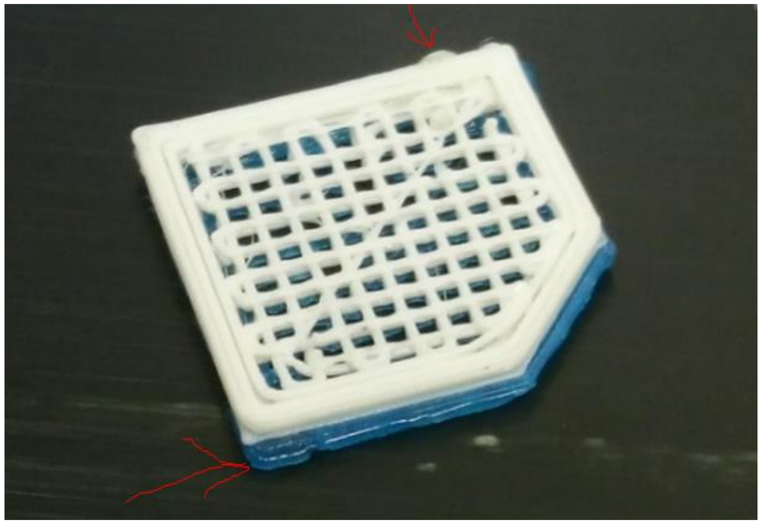


Figure 4-15 Ooze at interface of a stacked structure

Among the few solutions to minimize oozing, is to enable module retraction in Slic3r. When retraction is enabled, the extruder stepper pulls back some of the filament to reduce the backpressure in the nozzle. This temporarily limits oozing, but if the hot nozzle is idled for a long period, the heat expands the filament in the hot-end, which slowly oozes out of the nozzle due to slight increase in pressure and assisted by gravity and capillary action. Moreover, the low-pressure causes no extrusion until the backpressure rises back to push out the molten thermoplastic.

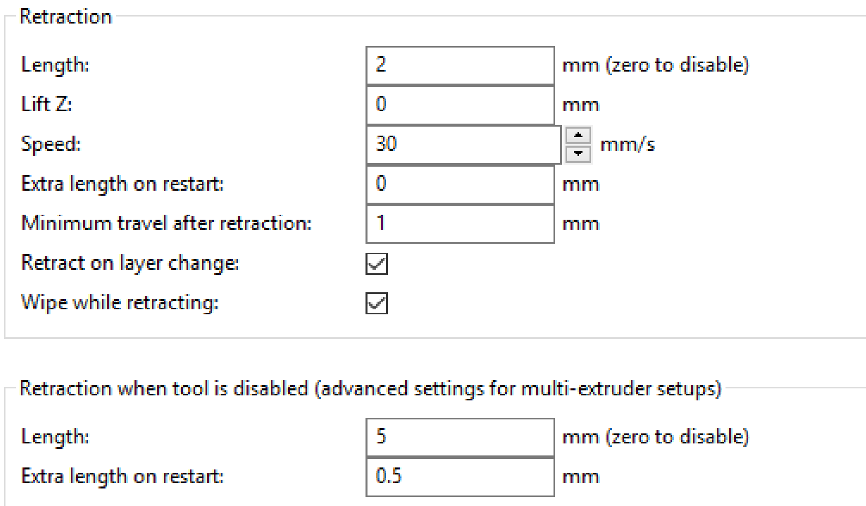


Figure 4-16 Retraction settings in Slic3r

An option in the retraction settings (Figure 4-16) called 'Extra length on restart' can be used to increase the backpressure before printing, but it provides limited control on how much to extrude. The value entered in this parameter is a static value that is constant irrespective of the amount of material oozed. The amount of material oozed is a function of temperature and idle time. The idle time varies with each toolchange, and the static value chosen for extra length on restart might be too low or too high. If it is too low, the extruder does not extrude until the pressure rises and if the length is too high, it causes excessive oozing. Figure 4-17 shows the effects of both conditions. The blue section was not completely printed due to insufficient extrusion, whereas white PLA was oozed and dragged onto the blue section. Slic3r has advance features including custom toolchange G-Code and temperature control for each module, but those features are insufficient for multi-modality systems.

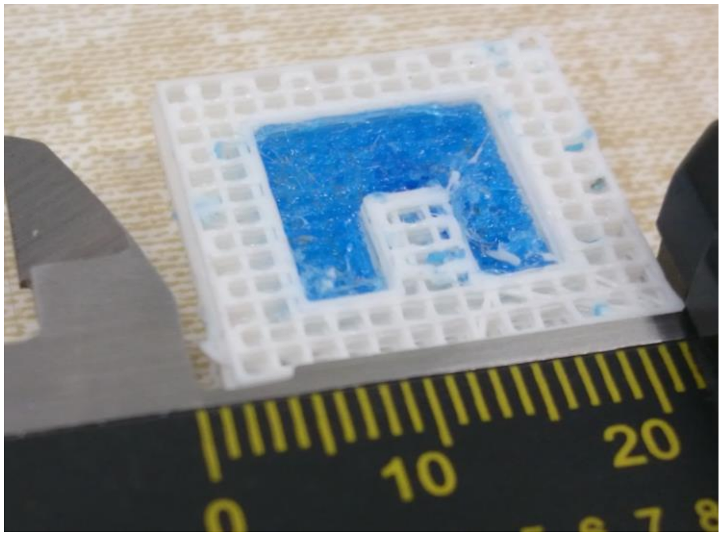


Figure 4-17 Insufficient extrusion due to low backpressure and oozing due to extra length at restart

Another solution to limit oozing is to turn off heating of the idle extruder by adding the cooldown command to the post-processing script after the pause G-Code command. The cooldown command is followed immediately by the heating command for the active extruder. To minimize the waiting time between cooldown and heating, 'M104' (do not wait for temperature) is used for cooling while 'M109' (wait for temperature) is used for heating. This allows the heating of active extruder, while the idle extruder is cooled down. A disadvantage of this method is that the cooling and heating of extruders take right above the printed part, which may be damaged during long exposure to heat from the hot extruders. The damage due to heat can be mitigated by using a set of move-away commands before cooldown and heating. Once the cooldown and heating procedures complete, the head will move back to print position after switching the module.

Moreover, the cooling and heating of extruders are processes that occur over a long waiting period. Over such large periods of idling, oozing may still occur while the extruder temperatures are over the melting temperature of their respective thermoplastic.

Lastly, the temperature based ooze control methods are not applicable to Viscous Extrusion (VE) module as they lack heating and temperature feedback. Retraction is the only method to limit ooze in VE. The amount of material to retract is material dependent. Material with very high viscosity usually does not need large amount of retraction while material with low viscosity needs larger retraction. The retraction process on VE must be a carefully controlled process to avoid introducing bubbles in the syringe. A small bubble at the base of the syringe can be allowed to form, but it needs to be flushed out prior to resuming the print.

While none of the above discussed methodologies were successful in combating the issues faced due to oozing, a combination of all of them can be used to limit the effects of material ooze. The post-processing script was further improved to include G-Code commands for retraction, move-away and cooldown to control oozing. After cooldown, the active extruder, before resuming the print process, was primed to restore printing backpressure. The script was developed while keeping the ‘Concept of Modularity’ as its development philosophy (Section 2.1). Several functions of Slic3r such as the custom toolchange G-Code commands are global, which means that they are applied to all modules. In multi-modality setups, each module must be independent and may need to have its own toolchange commands. When adding G-Code commands for performing tool change commands, the script improved to allow specific modules to be controlled by their own set of commands.

When the script finds a toolchange command ‘T#’, where ‘#’ is the tool number assigned, the script copies the G-Code from a specified file. This file, named ‘dump#’ is a regular text file that contains G-Code commands. For the FFF when the toolchange occurs, the head is moved away and the previous extruder is cooled down, while the current extruder is heated. Once the extruder is hot enough, some filament is pushed out to prime the nozzle before printing. The nozzle is then wiped on the print bed to remove any excess thermoplastic after which the extruder is raised slightly before it travels back to the print position. All these commands were stored in a text file named 'dump0'. The 'dump0' was used globally for all FFF modules, but this behavior can be modified if each FFF module needs a different set of commands. For VE modules 'dump1' was defined. Some example dump G-Code are presented in Appendix A.

An additional complication introduced due to moving the extruder away for module change was that the extruder would hit the build structure when returning to its printing position. Also, when the toolchange command was within a single layer, there was no immediate motion G-Code that would set the Z height. This caused the module to print at the wrong height.

An additional function was declared within the awk-based script that kept track of the height of the module. The tracking was a live counter and was updated whenever a Z change would occur. To keep a better track of Z-height and to restore the height after toolchange, the function added the Z-height after every move. As a side-effect this increased the number of bytes that were sent to the printer, but the increase was negligible and there was no noticeable change in performance of the printer. After the toolchange command, the function would add 5mm to the last known Z-height. This prevents the module from colliding with the print or dragging over the print surface. These parameters can be modified for individual modules.

4.3 Creating Heterogeneous Structures using a Design Level Approach

The design of the structure plays an important role in manufacturing an object using multiple materials and multiple geometries. A solid object maybe sliced using Slic3r, but the interior will be homogenous in composition. Such structures are perfect when the internal structure does not matter and exist only to support the top, bottom and the perimeter of the part. When creating heterogeneous structures, common slicing software lack the ability to define the interior structure to be heterogeneous.

One method to do this is to slice the object multiple times with different settings and then merge them using post-processing scripts to create a heterogeneous structure. This shall be discussed in the next chapter. Another approach is to define the sub-structure within the design itself such that the Slic3r only generates toolpaths without making modifications based on its own algorithms. Such method can be called as ‘Design Level Approach’. This can also be used to create homogenous structures.

Design level approach builds up on the concept of Unit Cell. A ‘Unit Cell’ can be considered as the smallest building block, whose repetition in space produces a structure whose geometry and arrangement defines the characteristic symmetry of the structure. A homogenous structure has just one unit cell whose characteristics are visible throughout the structure. For heterogeneous structures, there maybe two or more than two unit cells, each occupying a different region of the structure. Each region of the structure has the characteristics of the unit cell occupying that region. Figure 4-18 shows two homogenous structures being combined to create a heterogeneous structure.

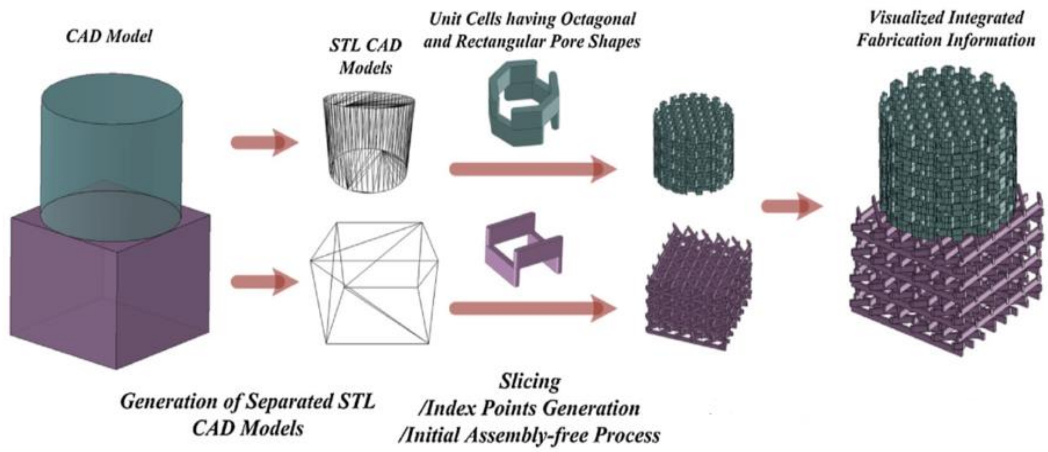


Figure 4-18 Two structures with homogenous unit cells combined to obtain a heterogeneous structure [34]

The models for this research were designed in SolidWorks 2015. Prior to modeling an object, certain parameters need to be determined. Some parameters are hardware dependent while others can be modified. Some of those parameters are:

1. Bed size – The bed size determines the maximum structure that can be constructed. This is hardware dependent.
2. Type of unit cell – The geometry of the unit cell governs the overall characteristics of the construct.
3. Homogenous or Heterogeneous – Homogenous structures have a single unit cell. Heterogeneous structure has multiple unit cells.
4. Distance between two adjacent cells – The distance between two adjacent cells is governed by the nozzle diameter. It's recommended to keep this parameter to be a multiple of nozzle diameter.
5. Layer Height – The unit layer needs to be as thick as the thickest extruded strand and is as defined layer height. Normally it is 0.2mm on the CMMB.

The unit cell is usually selected to be two or more shapes, which may or may not be the same. These shapes maybe referred to as sub-cells. Each sub-cell can be independent of each other in terms of shape and size. A minimum of two sub-cells are needed to complete a unit cell, as it will tell the relative position of two cells across the unit layer. The positioning of individual sub-cells is important and can be limited due to shape and hardware. With these parameters, different geometries were generated with various porosities and pore geometry.

The process to develop a structure using this method involves creating a two-dimensional shape which defines the structural and porous properties of the construct. The minimum feature size is always equal to the nozzle diameter. After the unit cell is prepared, it is patterned across the X and Y axis. After the unit cell is prepared, it is patterned across the X and Y axis. The pattern created gives the profile of a 'unit layer'. The patterning of each unit cell is performed such that the distance between two hexagons is always the nozzle diameter or the multiple of it and the final area is slightly larger than the required structure. Next a perimeter is created of required dimensions connecting either the center of hexagons or the outer edges. The unit layer profile is extruded in the Z direction to create a unit layer. The generated layer can be used as a building block for the structure. Each layer or a group of layers can be different height or have a different layer profile with varied settings such as infill type and density. The minimum layer height is equal to the layer height defined in the Slic3r. A unit layer is saved as an individual part (.SLDPRT) in SolidWorks. Multiple unit layers can be imported in an empty project and joined together using 'Mates'. The final assembly must be fully defined and exported as AMF using steps defined in section 4.2.

In Figure 4-19, three of the extruded unit layers are assembled on top of one another to create a homogenous structure having total thickness of 0.60 mm. As the file exported is in AMF, each sub-structure was assigned an individual extruder, such that each layer can have individual properties.

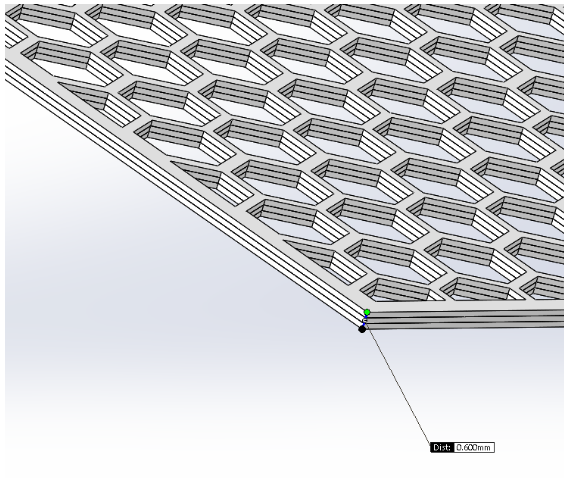


Figure 4-19 A structure of thickness 0.60mm created by combining unit layers

The AMF can now be imported into Repetier-Host and then printed. In the current example (Figure 4-20), each layer was assigned an extruder/process such that Layers 1 and 3 were assigned FFF (blue) and Layer 2 was assigned FFF (white).

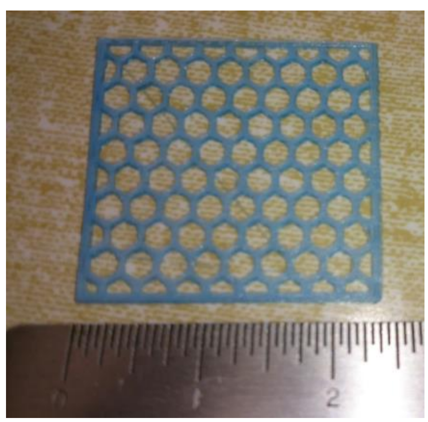


Figure 4-20 Homogenous hexagon structure with three layers manufactured on the CMMB

A unit cell, consisting of sub-cells of varying shapes or sizes may not create a heterogeneous structure. This is because the complete unit cell, along with its sub-cells, is patterned as a single entity across the plane which imparts a homogenous nature to the plane.

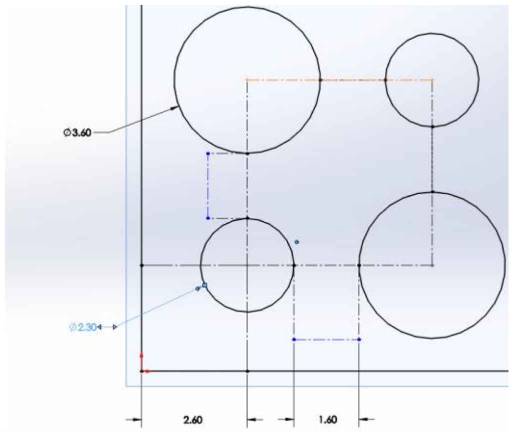


Figure 4-21 Variable porosity homogenous unit cell

In this type of unit cell as shown in Figure 4-21, each unit cell is flipped and mirrored such that it has four circular voids. This unit cell example demonstrates the importance of spacing between each cell unit. On the CMMB, it was determined that for the unit cell to be structurally strong, the distance between unit cells must be more than the nozzle diameter. The distance between two adjacent voids is chosen to be 4 times the nozzle diameter. As this is a hardware dependent parameter the minimum can be identified through experimentation for each printer and adjusted appropriately. Next, the unit cell was patterned across X and Y plane to obtain a profile of the unit layer which was later extruded to form the unit layer.

Multiple unit layers can then be stacked in an assembly and exported to AMF. The extrusion width is selected as half in the Slic3r settings. This is important for voids which have curved perimeter. The Slic3r, when creating the toolpath lays down the material in rectilinear alignment. This leaves gaps on the corners which are not filled in with the extruded material resulting in poor structural integrity. The extrusion width parameter is chosen to be the same as layer height to maintain structural integrity. Figure 4-22 shows a component created using a unit cell with varied porosity.

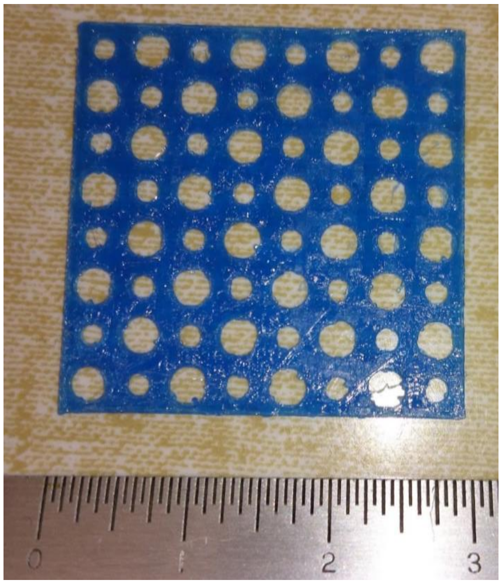


Figure 4-22 Homogeneous Structure with voids of varied porosity

Heterogeneous structures can be created with design level approach by selecting two unit cells and placing them at two different regions of the structure.

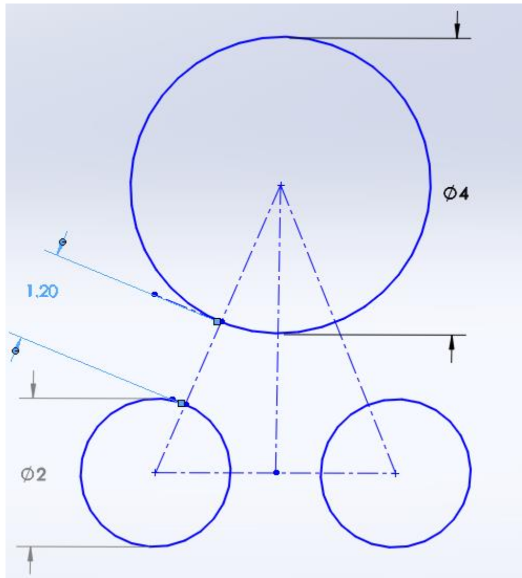


Figure 4-23 Heterogeneous unit cell with different sized circular voids

Figure 4-23 shows the interface where the two heterogeneous sections of the unit cell meet. The tangential distance between cells was chosen to be three times the nozzle diameter to increase structural integrity. Lower values resulted in breakage, although if the tangential distance is kept at double the nozzle diameter and extrusion width reduced to 50% of nozzle diameter, the structure did not break, but was weak and easily damaged .

These parameters are hardware dependent, and vary from hardware to hardware depending on nozzle diameter and layer height. On the CMMB with 0.4mm nozzle, the tangential distance of 1.2 mm with minimum extrusion width of 0.2 mm were defined.

Once the unit cells were chosen and parameters selected, they were patterned in their allocated regions across in the X-Y plane to obtain a unit layer profile. The unit layer profile was then extruded to form a unit layer.

Figure 4-24 shows how multiple unit layers can be combined in an assembly, exported in AMF and then fabricated.

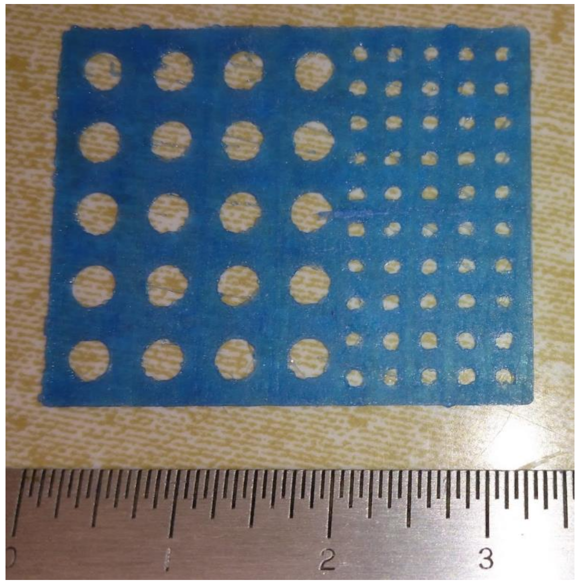


Figure 4-24 Heterogeneous component with different sized circular voids

Components with different geometries can also be created that have multiple geometries and material (Figure 4-25). In this example, two unit cells of rectilinear shape are chosen and patterned within their regions on the unit layer. The intersection of adjacent unit cell must be shared by both unit cells. This is needed to improve adhesion of the unit cells. An 1:1 ratio can be assumed safe for similar materials, but could be changed to other ratios for dissimilar materials or geometries as needed or desired.

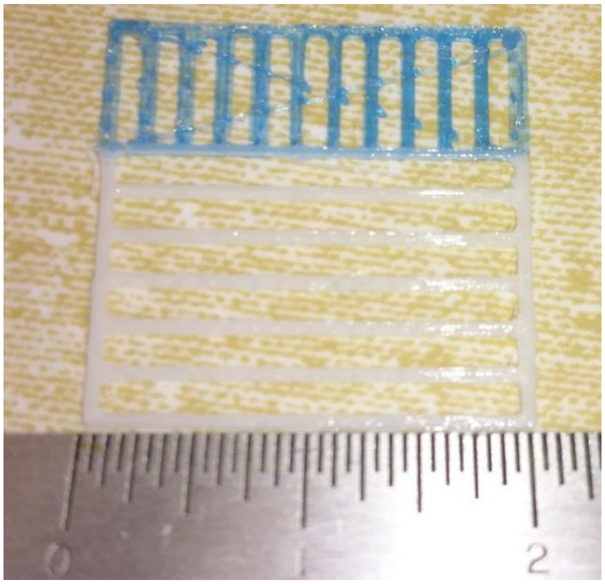


Figure 4-25 Heterogeneous structure with rectilinear unit layer containing heterogeneous materials and geometry

4.4 Creating Heterogeneous Components by post-processing

When creating structures using Slic3r and Repetier-Host, only homogenous structures are possible since Slic3r lacks the ability to define the heterogeneous structures. The exception to this is if the structures are designed using design level methodologies, then the Slic3r algorithm has no effect on the design except to generate the toolpath. As compared to design level methods, where the structure is constructed from the beginning based on unit cell parameters, the post processing algorithm allows us to modify homogenous solid structures into heterogeneous structures. This is faster compared to design level method as this relies on Slic3r's capability to create infills, but does not provide full control over features.

A post-processing algorithm, named zcomb, was developed to create a heterogeneous structure using conventional designs that have not been created using the design level methodologies. This involves slicing a structure multiple times with desired settings and then selectively but automatically combine the generated G-Codes such that desired regions have the desired structural properties. The process flow of the post-processing script is presented in Figure 4-26. The algorithm was written in C and the compiled executable is called using a batch script or from a command line interface. The batch script includes the set of G-Codes to be combined and defined heights at which the different G-Code are to be combined along with operator symbols. Operator symbols tell the program if the file should be copied completely or if it should be copied partially at the defined heights. The '=' operator symbol is used to copy complete G-Code and used for copying the start and end G-Code. It can also be used to completely copy other G-Codes as well. The '[' and ']' operator symbols are used to define the G-Code that are to be partially copied along with the start and end heights.

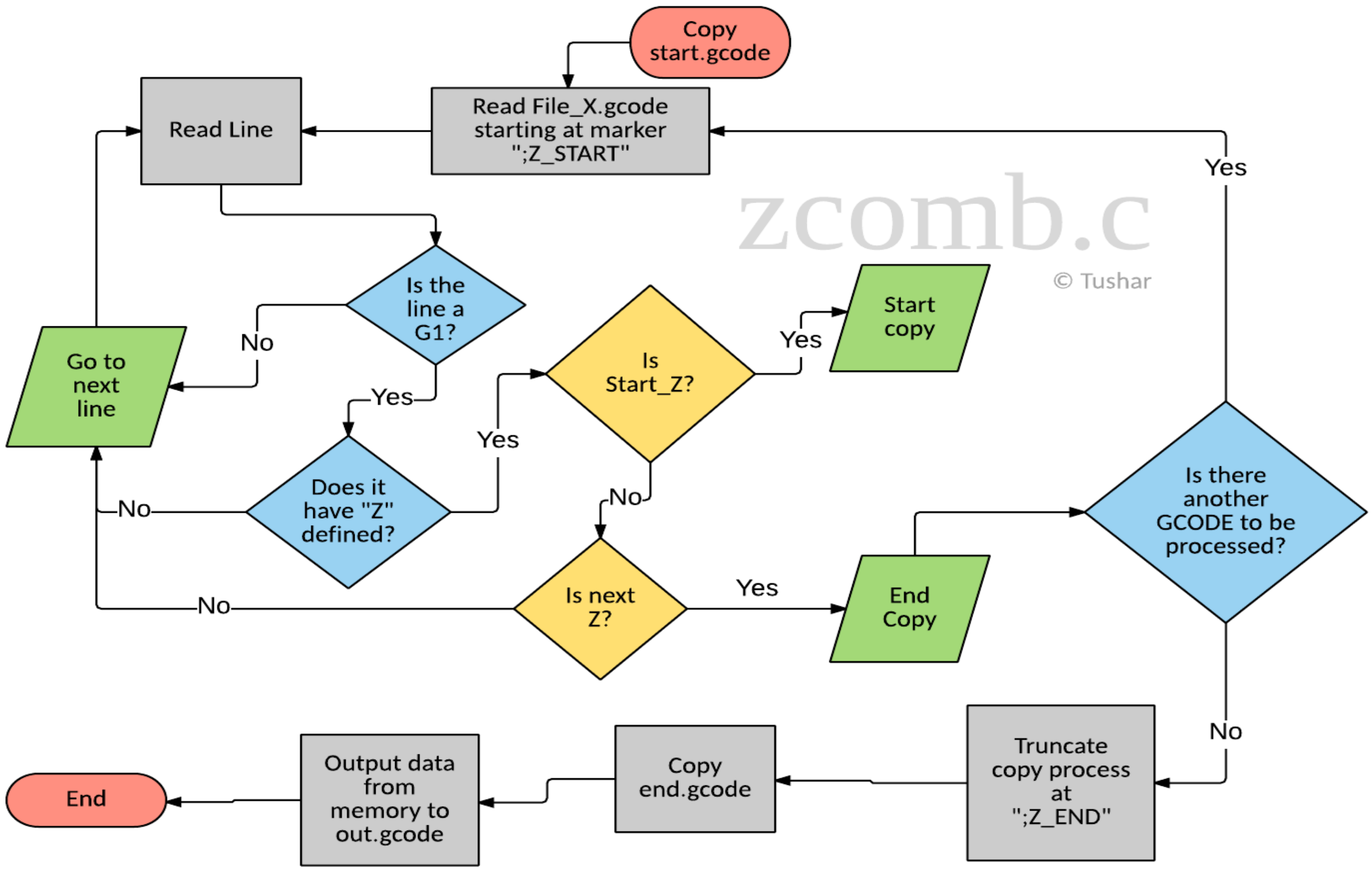
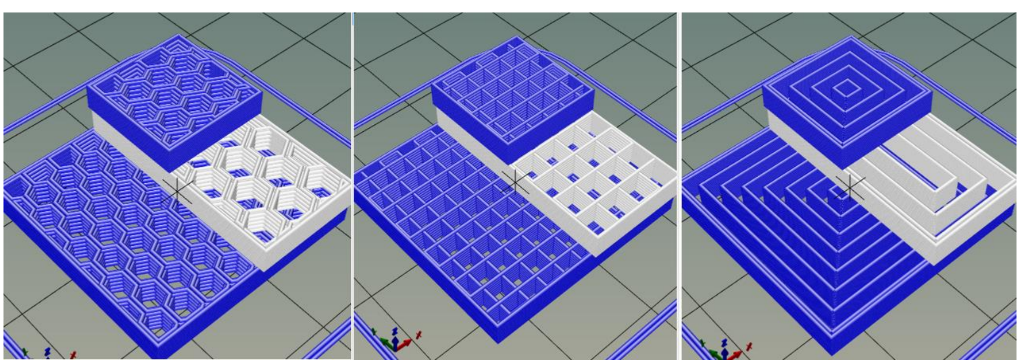


Figure 4-26 Flowchart to show the working of 'zcomb' post-processing algorithm

The algorithm starts by copying the start G-Code which contains startup routines such as homing of axes, heating the modules and setting units etc. This needs to be declared as the algorithm ignores the start and end G-Code specified in Slic3r. Next, the first G-Code file is opened and the algorithm looks for the START and the END markers. The algorithm copies G-Codes that are between the markers. This prevents duplication of start and end G-Code. The copy process continues up to the defined Z height at which the copy process is stopped and the algorithm imports the next G-Code. This process repeats until all the G-Code are processed for the defined heights. Finally, the end G-Code is copied. The complete copying process occurs in the application memory, and when the process is completed, the copied G-Code is routed using '>' operator to an output file.

The algorithm can be explained using the following example (Figure 4-27) shows a structure that has been sliced three times using different settings. In first slicing, a honeycomb pattern was used as the infill type with 30% infill density (Figure 4-27(a)). The G-Code is saved as '1.gcode'. A rectilinear pattern was used as the infill pattern with 20% infill density for the second G-Code. This was saved as '2.gcode' (Figure 4-27(b)). Finally the model was sliced the third time using concentric infill with 30% infill density and saved as '3.gcode' (Figure 4-27(c)).



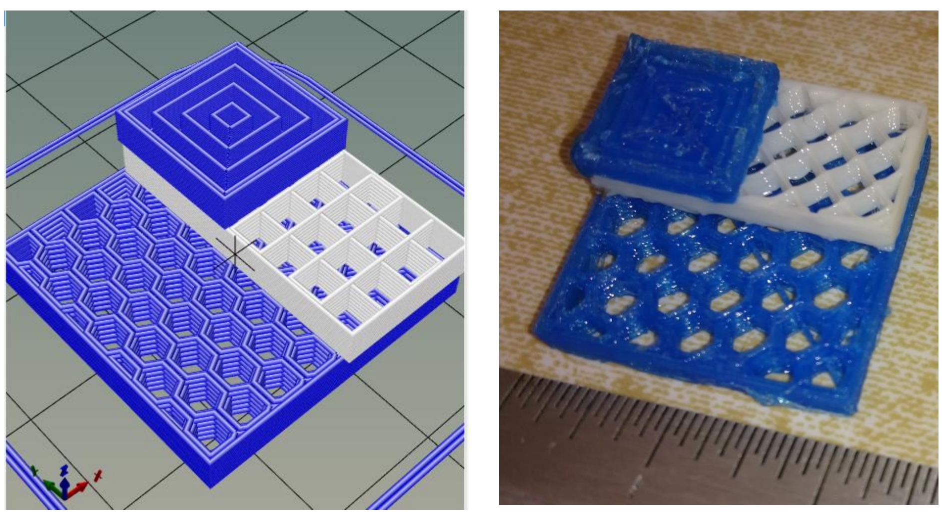
(a) Honeycomb – 30% (b) Rectilinear – 20% (c) Concentric – 30%

Figure 4-27 Homogenous structures sliced using Slic3r.

All the saved G-Code are placed in the same folder as the zcomb executable. In command prompt, the following command is used to combine G-Codes as required

```
zcomb = start.gcode [1.gcode 3.0 2.gcode 6.0 3.gcode] = end.gcode
>out.gcode.
```

The ‘start.gcode’ and ‘end.gcode’ are copied as a whole, while ‘1.gcode’ is copied between 0 mm and 3 mm; ‘2.gcode’ is copied between 3.0 mm and 6 mm; and ‘3.gcode’ is copied from 6 mm to the end of file. The final G-Code is exported to out.gcode and fabricated as presented in Figure 4-28.



(a) (b)

Figure 4-28 Heterogeneous structure combined using zcomb as previewed in Repetier-Host G-Code viewer and (b) final fabricated structure on CMMB

Some limitations of the current algorithm are that the combining of G-Code can only take place along the Z axis. An algorithm that will combine within the same layer is being worked upon.

Chapter 4 begins with the discussion on the need of heterogeneous functional components and a general manufacturing approach is discussed to create such components. The discussion covers how the properties of AMF file format can be used to define multiple objects and assigned materials. Next, the limitations of current software are covered, including issues like oozing and toolchange commands. A post-processing script is introduced that is aimed at resolving those issues. The chapter further follows the development of the post-processing script before discusses two methodologies of creating heterogeneous components using unit cells or post-processing.

Chapter 5

Conclusion and Future Work

Additive Manufacturing processes have been developing and improving continuously and provide us with a platform to create structures that exhibit various properties. Current AM machines and processes are primarily restricted to create homogenous structures that have single material properties. Creating functional components such as biological scaffolds or aircraft parts usually require heterogeneous structures that are fabricated using multiple materials and geometry which impart unique properties to the structure. Using design level methodologies and post-processing methodologies (Chapter 4), additive processes can be brought together to create fully functional prototypes. These methodologies, developed and tested on the Custom Multi-Modality BioPrinter (CMMB) at the MARS Lab at The University of Texas at Arlington bring together additive manufacturing processes such as multiple Fused Filament Fabrication (FFF) with different materials and Viscous Extrusion (VE) to create heterogeneous structures. Some important accomplishments of the research include:

1. The ability to control each module individually or in combination of multiple processes.
2. To define design level methodologies for Additive Manufacturing to create homogenous and heterogeneous structures that have all structural features defined
3. To be able to create fully functional structures fabricated using heterogeneous geometries and materials.
4. The manufacturing processes use existing technologies which are open source and can be modified and improved as required.

Next steps in this research would include improving the scripts to be able to include seamless heterogeneous structures in X and Y plane in addition to Z plane. An elementary post-processing algorithm (xycomb.c) has been prepared to create structures which have heterogeneous structures within a layer that uses toolchange commands to determine the position of geometry and materials. As the primary motivation for the research was to create scaffolds for cell growth and proliferation, the multi-modality additive manufacturing processes needs to be further developed taking into account the scaffold's structural and porous properties. Topology optimization with FEA analysis can be performed for defining the structural and porous properties which can be used to create scaffolds using FFF modules and embed cells using either VE or Inkjet or a combination of both. For the above created post-processing scripts to be the most beneficial, it would be best to have them merged with a toolpath creation software (eg. Slic3r). Lastly, the newer 3MF format that was developed to bring 3D printing to masses can be used to integrate post-processing techniques so as to create heterogeneous structures without any additional steps.

Appendix A

Sample Toolchange G-Code

Sample G-Code for FFF Module:

Filename: dump00

```
; Moves Z up by 4mm
G91
G1 Z4.000 F6000
G90
; Move to dump area
G1 X130.000 Y0 Z0.2000 F6000
;;; Change Temperatures
M104 S140 T0
M109 S190 T1
;;;;Dump
G92 E0
G1 E10.000
G1 E-0.005
G1 X130.000 Y50.000 Z0.200
G1 X130.000 Y55.000 Z0.200
G1 X140.000 Y50.000 Z0.200
G1 X145.000 Y80.000 Z0.200
G1 F6000
```

Sample G-Code for VE Module:

Filename: dump30

```
; Moves Z up by 4mm
G91
G1 Z4.000 F6000
G90
; Move to dump area and dump
G1 X150.000 Y30 Z0.1000 F6000
G92 I0
G1 I0.050
G1 I-0.001
G1 F6000
```

References

[1] X. Yan and P. Gu, "A review of rapid prototyping technologies and systems," *CAD Computer Aided Design,* vol. 28, no. 4, pp. 307-318, 1996.

[2] I. Gibson, D. W. Rosen and B. Stucker, in *Additive Manufacturing Technologies*, USA, Springer Science + Business Media, 2010.

[3] "3D Printing Technologies: Stereolithography," I.Materialize, [Online]. Available: https://i.materialise.com/blog/an-intro-to-our-3d-printing-technologies-stereolithography.

[4] Zureks, "Wikipedia," [Online]. Available: https://commons.wikimedia.org/wiki/File:FDM_by_Zureks.png.

[5] "3DPrinting," [Online]. Available: http://3dprinting.com/what-is-3d-printing/.

[6] "RepRap Project," Wikimedia Foundation, Inc, 13 October 2015. [Online]. Available: http://reprap.org/wiki/Main_Page.

[7] S. Song, J. Choi, Y.-D. Park, J. Lee, S. Y. Hong and K. Sun, "A Three-dimensional Bioprinting System For Use With A Hydrogel-based Biomaterial And Printing Parameter Characterization," *Artificial Organs,* vol. 34, no. 11, pp. 1044-8, 2010.

[8] S. Khalil, J. Nam, A. Darling and W. Sun, "Multi-nozzle Biopolymer Deposition For Freeform Fabrication Of Tissue Construct," in *14th Interdisciplinary Research Conference on Biomaterials*, Limoges, France, 2004.

[9] T. Xu, K. K. W. Binder, M. Z. M. Albanna, D. Dice, W. Zhao, J. J. Yoo and A. Atala, "“Hybrid Printing Of Mechanically And Biologically Improved Constructs For Cartilage Tissue Engineering Applications," *Biofabrication,* vol. 5, no. 1, p. 015001, 2013.

[10] P. Ravi, P. S. Shiakolas, T. Welch, T. Saini, A. K. Batra and K. Guleserian, "ON THE CAPABILITIES OF A MULTI-MODALITY 3D BIOPRINTER FOR CUSTOMIZED BIOMEDICAL DEVICES," in *Proceedings of the ASME 2015 International Mechanical Engineering Congress and Exposition* , Houston, Texas, USA, 2015.

[11] [Online]. Available: http://www.reprapdiscount.com/electronics/33-rumba-basic-board.html.

[12] S. V. Murphy and A. Atala, "3D bioprinting of tissues and organs," *Nat. Biotechnol.,* vol. 32, no. 8, pp. 773-785, 2014.

[13] S. Yamaguchi, A. Ueno, Y. Akiyama and K. Morishima, "Cell patterining through inkjet printing of one cell per droplet," *Biofabrication,* vol. 66, no. 3, pp. 911-917, 2012.

[14] P. Ravi, P. S. Shiakolas, A. Batra, J. Oberg and S. Faizee, "ON THE DEVELOPMENT OF A MODULAR 3D BIOPRINTER FOR RESEARCH IN BIOMEDICAL DEVICE FABRICATION," in *Proceedings of ASME 2015 International Mechanical Engineering Congress and Exposition*, Houston, Texas, USA, 2015.

[15] B. K. Parka, D. Kima, S. Jeonga and J. Moon, "Direct writing of copper conductive patterns by ink-jet printing," in *Proceedings of Symposium I on Thin Films for Large Area Electronics EMRS 2007 Conference — EMRS 2006 - Symposium I*, 2007.

[16] P. Calvert, "Inkjet Printing for Materials and Devices," *Chem. Mater,* vol. 13, no. 10, pp. 3299-3305, 2001.

[17] "RUMBA - RepRapWiki," RepRapWiki, 11 September 2015. [Online]. Available: http://reprap.org/wiki/RUMBA. [Accessed 13 October 2015].

[18] "Slic3r - RepRapWiki," RepRap Project, 08 10 2015. [Online]. Available: http://reprap.org/wiki/Slic3r. [Accessed 29 10 2015].

[19] "Slic3r," [Online]. Available: http://slic3r.org/about.

[20] "Main Page - MarlinFirmware," [Online]. Available: https://github.com/MarlinFirmware/Marlin/wiki/Main-Page. [Accessed 11 10 2015].

[21] C. Trinh, "chtrinh/Marlin," Github, March 2014. [Online]. Available: https://github.com/chtrinh/Marlin/tree/bioprinter. [Accessed June 2015].

[22] E. Weisstein, "Triangulation -- from Wolfram MathWorld," Wolfram Web Resources - MathWorld Team, [Online]. Available: http://mathworld.wolfram.com/Triangulation.html.

[23] N. M. Amato, M. T. Goodrich and E. A. and Ramos, "A Randomized Algorithm for Triangulating a Simple Polygon in Linear Time," *Discrete Computational Geometry,* no. 26, pp. 245-265, 2001.

[24] I. 3D Systems, "StereoLithography Interface Specification". July 1988.

[25] I. 3D Systems, "StereoLithography Interface Specification, 3D Systems, Inc.". October 1989.

[26] I. 3D Systems, "SLC File Specification". 1994.

[27] LaurensvanLieshout, "Wikimedia Commons, the free media repository," 07 Marcg 2009. [Online]. Available: https://en.wikipedia.org/wiki/File:The_differences_between_CAD_and_STL_Models.svg. [Accessed November 2015].

[28] Fabbers.com, "The STL Format: Standard Data Format for Fabbers," [Online]. Available: http://www.fabbers.com/tech/STL_Format.

[29] maxim_k, "Coreldraw Community," [Online]. Available: https://community.coreldraw.com/talk/technical_graphics_products/f/642/t/49016.

[30] "3MF | Overview," 2015. [Online]. Available: http://3mf.io/about-us/overview/.

[31] "3D Manufacturing Format Specification and Reference Guide," 2015.

[32] A. S. f. T. a. Materials, "Standard Specification for Additive Manufacturing File Format (AMF)," 2011. [Online]. Available: https://amf.wikispaces.com/file/view/AMF_V0.48.pdf. [Accessed 2015].

[33] J. Li, K. Kim, S. Roohani-Esfahani, J. Guo and D. Kaplan, "A biphasic scaffold based on silk and bioactive ceramic with stratified properties for osteochondral tissue generation," *Biol Med,* vol. 3, no. 26, pp. 5361-5376, 2015.

[34] H.-W. Kang, H. Jeong, T.-Y. Kang, Y.-J. Seol and W. C. Dong, "Unit cell-based computer-aided manufacturing system for tissue engineering," *Biofabrication,* vol. 4, no. 1, p. 015005, 2012.

[35] R. P. Lanza, R. Langer and W. L. Chick, "Principles of Tissue Engineering," *Austin, TX: Academic,* vol. 405, no. 27, 1997.

[36] S. H. Oh, I. K. Park, J. M. Kim and J. H. Lee, "In vitro and in vivo characteristics of PCL scaffolds with pore size gradient fabricated by a centrifugation method," *Biomaterials,* vol. 28, pp. 1664-71, 2007.

[37] C. E. J. George, D. P. Glenn, C. W. Simon, E. M. J. John and S. S. Michael, "Guidance of engineered tissue collagen orientation by large-scale scaffold microstructures," *J. Biomech,* vol. 39, pp. 1819-31, 2006.

[38] J. Malda, T. B. F. Woodfield, F. van der Vloodt, F. K. Kooy, D. E. Martens, J. Tramper, C. A. van Blitterswijk and R. J, "The effect of PEGT/PBT scaffold architecture on oxygen gradients in tissue engineered cartilaginous constructs," *Biomaterials,* vol. 25, pp. 5773-80, 2004.

[39] S. Miot, T. B. F. Woodfield, A. U. Daniels, R. Suetterlin, I. Peterschmitt, M. Heberer, C. A. van Blitterswijk, R. J and I. Martin, "Effects of scaffold composition and architecture on human nasal chondrocyte redifferentiation and cartilaginous matrix deposition," *Biomaterials,* vol. 26, pp. 2479-89, 2005.

[40] C. Y. Lin, K. N and S. J. Hollister, "A novel method for biomaterial scaffold internal architecture design to match bone elastic properties with desired porosity," *Biomechanicals,* vol. 37, pp. 623-36, 2004.

[41] W. Sun, F. Lin and X. Hu, "Computer-aided design and modeling of composite unit cells," *Compos. Sci. Technol.,* vol. 61, pp. 289-99, 201.

[42] E. J. Anderson and M. L. Knothe Tate, "Design of tissue engineering scaffolds as delivery devices for mechanical and mechanically modulated signals," *Tissue Engineering,* no. 13, pp. 2525-38, 2007.

[43] "The awk Language," [Online]. Available: https://www.gnu.org/software/gawk/manual/html_node/Getting-Started.html#Getting-Started.

[44] "Solidworks 3D CAD Tutorial: Using 3D Modeling Software for 3D Printing," Sculpteo, [Online]. Available: http://www.sculpteo.com/en/tutorial/prepare-your-model-3d-printing-solidworks/. [Accessed 28 10 2015].

[45] I. E. Moyer, "CoreXY | Cartesian Motion," 2012. [Online]. Available: http://corexy.com/theory.html.

[46] Materialgeeza, "SLS system schematic," May 2008. [Online]. Available: https://commons.wikimedia.org/wiki/File:Selective_laser_melting_system_schematic.jpg.

[47] LaurensvanLieshout, "Laminated object manufacturing," [Online]. Available: https://en.wikipedia.org/wiki/Laminated_object_manufacturing#/media/File:Laminated_object_manufacturing.png.

Biographical Information

Tushar Saini has received his Bachelor of Engineering degree in Aeronautical Engineering from the Hindustan University, Chennai, India. He has completed his Masters in Mechanical Engineering from the University of Texas at Arlington, USA in December 2015.

Tushar's interests lie in manufacturing, control systems and software development. He has experience in programming and deploying single board computers and currently works as a Special Projects Graduate Research Assistant at FabLab at the University of Texas at Arlington.