EXPLORING THE BENEFITS AND LIMITATIONS OF MULTI-AXIS 3D PRINTING FOR IMPROVED PART QUALITY AND REDUCED WASTE

J.A. ANDERSONS

SUPERVISOR:
Dr. Ir. T.H.J. Vaneker

THESIS ASSESSMENT BOARD:
Dr. Ir. T.H.J. Vaneker (chair)
Dr. Ir. D. Jafari
Ir. I. Lutters Weustink

Document number: DPM2033
Date of colloquium: 31st of August 2023

EXPLORING THE BENEFITS AND LIMITATIONS OF MULTI-AXIS 3D PRINTING FOR IMPROVED PART QUALITY AND REDUCED WASTE

J.A. ANDERSONS

Thesis submitted to the Faculty of Engineering Technology of the University of Twente in partial fulfilment of the requirements for the degree of Master of Science in Mechanical Engineering.

SUPERVISOR:

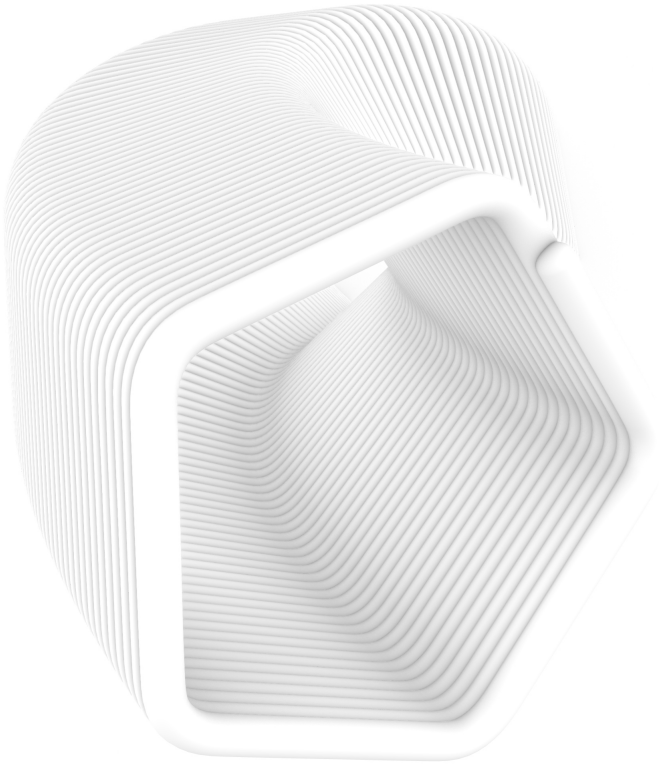
Dr. Ir. T.H.J. Vaneker

THESIS ASSESSMENT BOARD:

Dr. Ir. T.H.J. Vaneker (chair)

Dr. Ir. D. Jafari

Ir. I. Lutters Weustink



ABSTRACT

A MULTI-AXIS APPROACH TO ADDITIVE MANUFACTURING breaks the paradigm of planar 3D printing, enabling support-free printing of overhanging geometry, improved surface finish of oblique surfaces, and controlled mechanical anisotropy. These advancements yield reduced material and labor waste by eliminating the need for sacrificial support structures, through diminished physical material usage and circumvention of labor-intensive post-processing. Furthermore, the overall quality of parts is elevated, mitigating aliasing in shallow-angle surfaces inherent to the stratification of discrete layers.

Despite its potential, the widespread adoption of this technology faces hurdles in the domains of software and mechanical design. Adding degrees of freedom necessitates more complex path-planning algorithms and an accurate kinematic system. In order to harness the benefits of multi-axis additive manufacturing and ensure their practical implementation, it is crucial to make this technology accessible to a wider audience, particularly considering the significant contributions and limited resources of citizen-scientists within the 3D printing community.

Therefore, this thesis investigates the accessibility of multi-axis additive manufacturing technology through the development and validation of an open-design, self-replicating, low-cost retrofit for consumer-grade desktop 3D printers. To maximize build volume and kinematic accuracy, a 2-axis print-head was designed. Path-planning algorithms were developed and implemented to successfully print sample parts, showcasing the printer's ability to overcome the limitations of conventional printers while avoiding new drawbacks. The sample parts encompassed various geometries and sizes, serving as evidence of the printer's capability to produce intricate, large-scale components with unsupported overhanging features and smooth oblique surfaces. The surface quality and accuracy of the parts were evaluated in comparison to equivalent parts printed using a typical 3-axis process.

The work intends to generate interest in multi-axis additive manufacturing within the vast citizen-scientist community of 3D printing, to accelerate further development in the field.

ACKNOWLEDGEMENTS

Firstly, many thanks to **Dr. Ir. T.H.J. Vaneker** for supervising this thesis assignment, and guiding me through the writing process! Thanks to **Dr. Ir. D. Jafari** for assisting me with the 3D scanning used for the characterization of dimensional accuracy. And thanks to **Erik de Vries** for assisting me with the confocal microscopy for surface characterization.

Thanks to my family for supporting me throughout this two-year period, which has resulted in both my academic and personal growth.

A special thanks to my peers at DroneTeam Twente, with whom I shared my daily workspace: **Gerard Bel Catala** for teaching me lots about FDM and sharing my frustrations with GH; **Rieks Kaiser**, **Tom Kluin**, **Ahmed Mahran**, and **Tristan Voors** - for being great discussion partners, particularly on Friday nights.

Finally, thanks to the online 3D printing community at /r/3Dprinting, for exhaustive feedback and encouragement along the way.

TABLE OF CONTENTS

APPENDICES

LIST OF ACRONYMS

AM Additive Manufacturing. 2–6, 10, 12–15, 19–22, 24, 31, 32, 38, 63, 83

BOM Bill of Materials. xvi, 47, 59

CAD Computer Aided Design. xiii, xiv, xvi, 9, 55, 58, 59, 63, 64, 74, 75, 84, 85, 87–90, 98–100

CFRP Continuous-Fiber Reinforced Polymer. 32

CNC Computer Numerical Controlled. 9, 14, 34, 42, 44, 46, 60, 64, 66, 76, 99

COTS commercial off-the-shelf. 19, 41, 48, 59, 99

DoF Degrees of Freedom. 9, 12, 18–20, 36, 43, 44, 63, 98

EE End Effector. 9, 20, 53, 63, 77, 78

FDM Fused Deposition Modeling. x, xi, xiii, 2, 6–15, 17–20, 22, 26–28, 31, 32, 36, 38, 39, 41, 42, 44–48, 51, 53–56, 60, 61, 63–65, 70, 72, 76, 77, 82–84, 86, 92, 93, 96–99, 101

FK Forward Kinematics. 77

G-code Geometric code. xii–xiv, 9, 14, 17, 34–36, 38, 56, 63–68, 70, 72–76, 78, 79, 81, 99–101, 113, 114

GH Grasshopper. xiii, 32, 68–72, 75, 82, 84, 99, 118

HH head-head. xvi, 20, 21, 38, 44, 46, 53, 98–100

IK Inverse Kinematics. vi, xiii, 14, 47, 53, 54, 63, 72, 76–80, 101

PID Proportional-Integral-Derivative. 8, 9, 76

PLA Polylactic acid. 46, 47

SLA Stereo-lithography. 2

SM Subtractive Manufacturing. 2–6

TT table-table. xvi, 20, 21, 23, 42–46, 53, 98–100

LIST OF FIGURES

LIST OF TABLES

Chapter 1

Introduction

OVER THE DIGITAL ERA, 3D printing has grown to be a household name. Now a ubiquitous technology accessible to the layperson, it is as close as we've ever been to the sci-fi dream of a consumer device capable of producing any object. Yet, there is still a long way to go. Among the advancements in this field is the current emergence of multi-axis printing, breaking the paradigm of planar printing. This thesis explores the boundaries of this direction in 3D printing technologies, to answer *why it is worthwhile*, and *how to do it*.

This chapter provides an overview of the underlying technologies explored in this thesis, a motivation for performing the work, and the research objective.

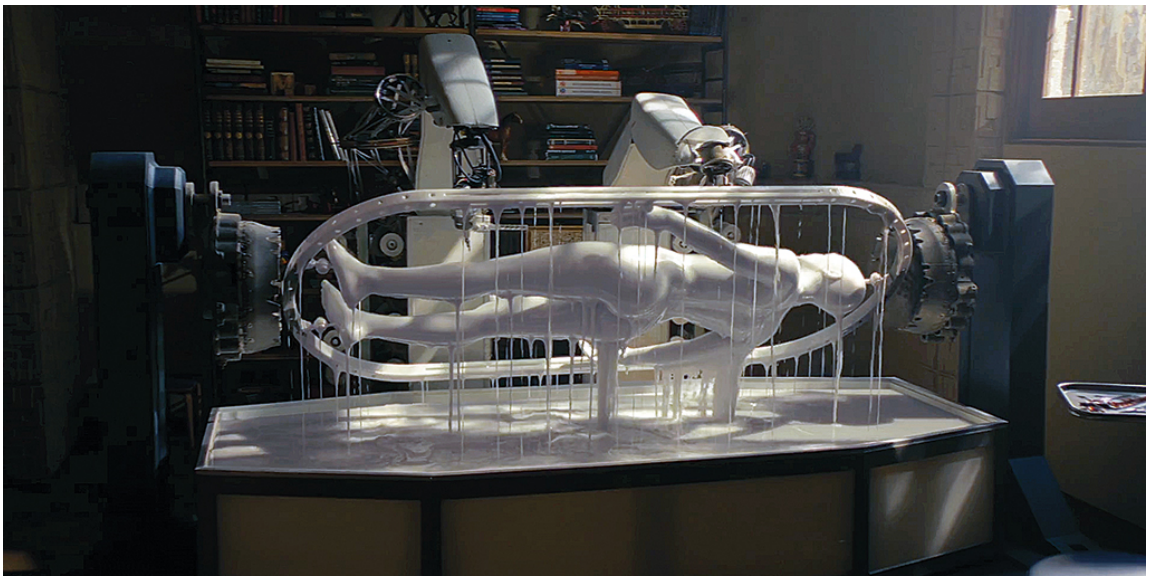


Figure 1.1: Fictional depiction of 3D printing an android in the TV series *Westworld*[1]

1.1 BACKGROUND

1.1.1 Additive Manufacturing

Additive Manufacturing (AM), colloquially referred to as 3D printing, is typically described as the process of fabricating a part layer by layer. [2] As the name implies, the processes involve building a part additively – by adding layers of material onto precedent layers. This is in contrast to Subtractive Manufacturing (SM) in which a part is built by subtracting or removing material from a bulk. The difference is shown schematically in Figure 1.2.[1]

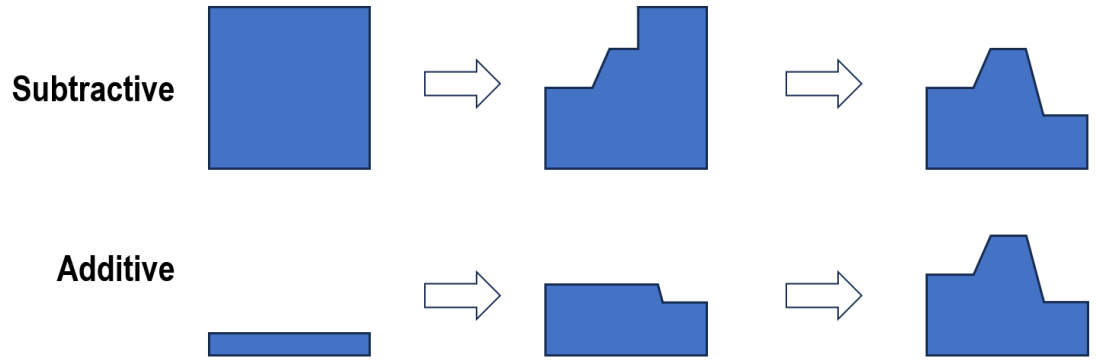


Figure 1.2: Fundamental working principles of Additive and Subtractive Manufacturing processes

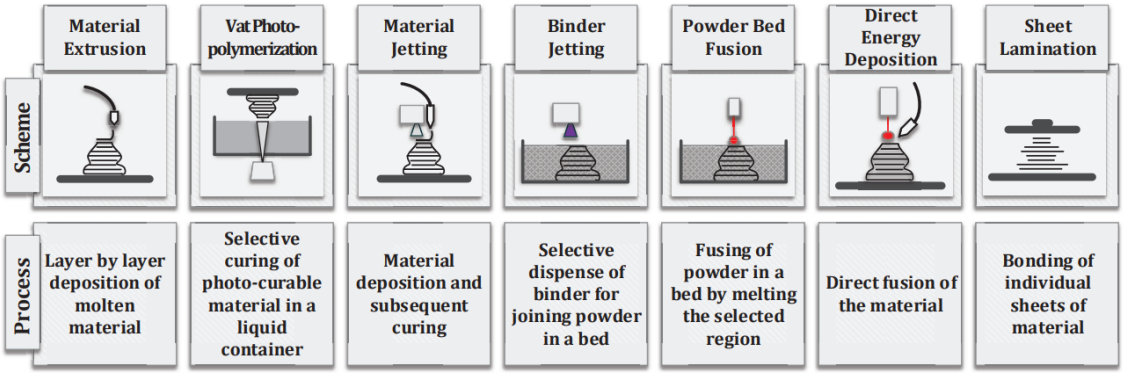


Figure 1.3: Additive Manufacturing processes[3]

Early development of AM technologies can be traced back to the 70's. The first patent for what could be considered an AM technology was granted in 1971, in which a stream of molten metal is deposited through a nozzle onto a carrier.[4] Developments on the Stereo-lithography (SLA) process happened in the 80's.[5] Within a few years several categories of AM were identified, and they are now compartmentalized into 7 general process categories,[6] an overview of which is shown in Figure 1.3. By the turn of the century, just about a thousand AM systems had been deployed. The AM space saw explosive growth with the expiry of Stratasys' patent on the FDM technology in 2009.[7] FDM has remained a mainstay in the AM industry, placed by several independent surveys as the most used process type.[8] In 2023 the market size of the entire 3D printing industry

[1] While many AM processes adhere to this definition, not all do, and in some cases it is not an inherent limitation.

is estimated at 17 billion USD with an annual growth rate of 13%.[8] What drives this growth in AM technology is its unique advantages:

Design flexibility AM processes typically offer a higher degree of flexibility in the design of parts, than with Subtractive Manufacturing (SM) or other traditional manufacturing methods. It is often said that for AM *"complexity is free"*, meaning that complex and intricate parts, like the one shown in Figure 1.4, are not necessarily any more expensive to manufacture. Whereas for SM or traditional processes that added complexity would require additional operations or more advanced tooling. Another aspect of design flexibility is material complexity - a wide range of materials can be used, ranging from engineering materials like thermoplastics, metals and fiber-reinforced polymers, to organic tissue and gels. Even combinations of materials in a single part.

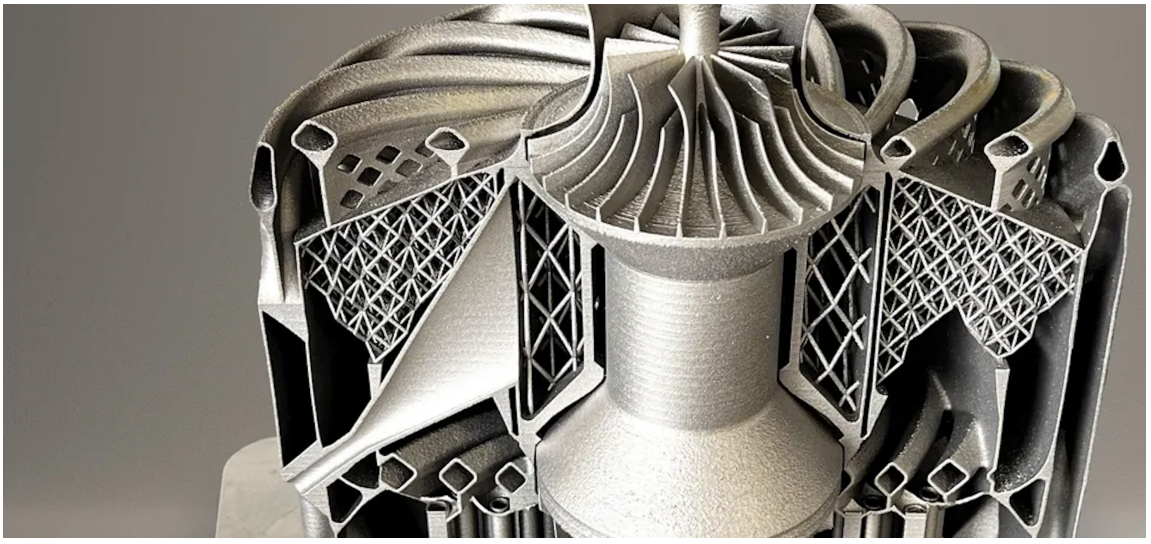


Figure 1.4: 3D printed micro-turbine illustrating the complexity and versatility enabled by AM[9]

Short production cycle AM typically has a short turnaround time, enabling shorter production and development cycles. To produce a part additively, no custom tooling, molds or stock material need to be procured, so a part can be quickly prepared for production in just minutes. This specific advantage makes it particularly appealing for rapid prototyping, as a design can be quickly iterated. But it has also proven useful in a mass-manufacturing environment. E.g. HVAC units on early Tesla Model Y cars were fitted with a 3D printed part, shown in Figure 1.5, due to a late revision. To not hold up production while waiting for the new injection molds, 3D printing was used as a stop-gap measure.

Customization Mass customization or ultra-personalization is the principle of being able to tailor each manufactured part. With traditional methods this would require additional tooling or processes, but with AM processes each part can be uniquely customized, to have certain dimensions, texture, or features. E.g. a shoe could be printed to perfectly fit a 3D scan of the client's foot,[11] shown in Figure 1.6.

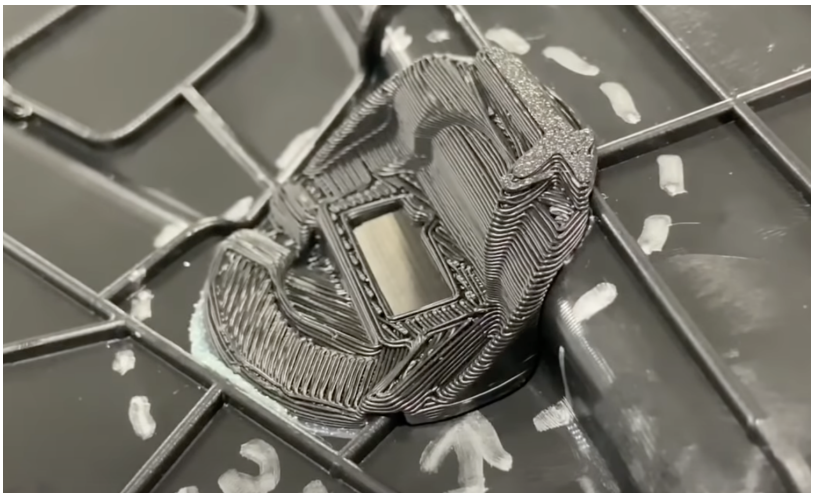


Figure 1.5: 3D printed part in early Tesla Model Y HVAC units[10]

Figure 1.6: Designer sneaker printed by Zellerfeld[12]

Reduced material use AM processes use roughly as much material as is needed to form the part, with the exception of sacrificial support structures (explained in Section 1.1.2.3). This makes them less wasteful than SM processes, where a larger bulk material is cut to shape, even though the scrap material may be recycled.

Low infrastructure requirements AM machines are compact and easy to set up in comparison to traditional manufacturing machines. Little to no additional infrastructure is needed, apart from power.[2] This can be effectively used to evolve an organization's supply chain, by relocating the manufacturing process closer to the end user. Reducing the lead time is valuable in industry, but in e.g. military use AM can allow the fabrication of mission-critical parts on the front line, as shown in Figure 1.7, and not have to keep an inventory of spares.

[2]Compared to SM requiring inventory of various size material stock, coolant systems incl. filtration and recirculation, tool library, waste management systems etc.

Accessibility The comparatively low barrier of entry to operating AM machines opens the technology up to new environments. In a development setting this can reduce the turnaround time by eliminating the need for a specialized machinist/technician. This also further contributes to supply chain evolution, as parts can be printed anywhere by personnel with minimal education. Crucially, it has led to the popularity of the technology in the hobbyist/maker/enthusiast space, which has driven new innovation.

Figure 1.7: Markforged X7 field printer used in a military environment[13]

Drawbacks

The aforementioned advantages of AM over traditional manufacturing practices make it an attractive alternative for various use cases. But most prominent of them is product development. Roughly two thirds of AM users state prototyping as their main use-case, whereas only around 20% state end-use parts.[8] This heavy sway toward development can be attributed to the shortcomings of AM processes:

Slow process Despite the short production cycle, the process time for AM is typically significantly longer. With traditional forming methods parts are done in seconds, and with SM processes the bulk material can be roughly machined using large cutters with high material removal rate. Meanwhile the layered nature of most AM processes has an inherent bottleneck in the speed of producing each discrete layer.

High material cost The material used for AM processes is typically more expensive than the equivalent material used for traditional processes - whether it is powder, thermoplastic filament, or photopolymer resin.[3] Considering the full manufacturing costs, AM has a low economic batch size, making it poorly suited for high-volume manufacturing.

[3] Gas atomization of metal powder and extrusion of thermoplastic filament adds cost to the raw material.

Inferior part quality AM processes generally can't produce parts to the same dimensional and geometric tolerances, and surface finish, as SM. If higher accuracy is needed additional processing steps must be included, like for formed parts. Additionally, many AM processes produce parts that are weaker than traditionally manufactured counterparts, due to poor inter-layer bonding.

Geometric limitations Design flexibility was previously mentioned as a key advantage, but it must be noted that, despite the high level of possible complexity, there are still limitations to the possible geometry that certain AM processes can produce. These limitations are highly dependent on the specific process.

1.1.2 Fused Filament Fabrication

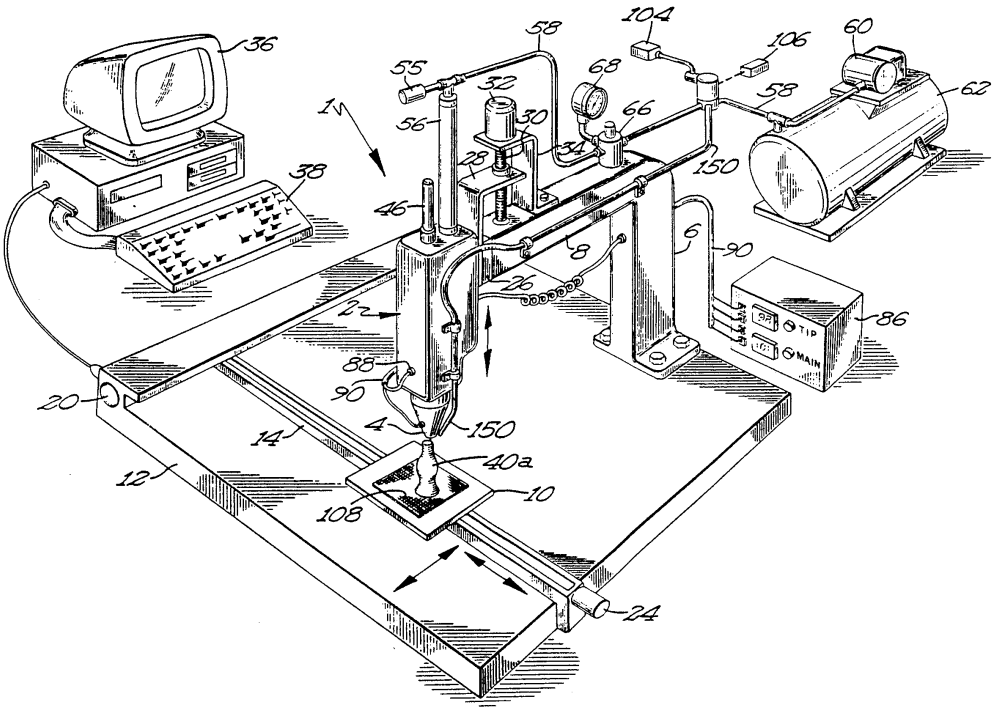


Figure 1.8: Schematic from the patent for Fused Deposition Modeling, invented by Crump[7]

Of the different categories of AM processes, Fused Deposition Modeling (FDM) stands out as the most commonly used.[8] The process was invented by S. Crump, and patented by Stratasys in 1992, shown in Figure 1.8.[7] The patent describes the printer as having a "movable dispensing head provided with a supply of material which solidifies at a predetermined temperature, and a base member, which are moved relative to each other along "X," "Y," and "Z" axes in a predetermined pattern to create three-dimensional objects by building up material discharged from the dispensing head onto the base member at a controlled rate."[7] Though the patent describes the use of a wider variety of materials such as "self-hardening waxes, thermoplastic resins, molten metals, two-

part epoxies, foaming plastics, and glass,"[7] FDM refers to specifically the use of thermoplastic filament as the material. Other similar systems are typically referred to as Material Extrusion.

1.1.2.1 Open design and citizen science in FDM

The expiration of Statasys' patent on FDM in 2009[7] kickstarted the rapid growth of the field, alongside the unexpected transformation from a niche manufacturing process to a mainstream consumer technology. One of the early projects to develop a low-cost 3D printer best exemplifies the attributes that have fueled innovation in the field. RepRap – a portmanteau of Replicating Rapid Prototyper - is an entirely open source project, with the initial goal of developing a 3D printer which can *self-replicate*.

The concept of self-replication is prevalent and intertwined with the open design principles. Contributors who design an improvement or alteration to their printer can themselves produce it using their existing printer. In an excellent example given by Bowyer, the second ever RepRap printer was suffering from issues due to an untensioned belt. A belt tensioner was then printed on that same printer to remedy the issue.[14] The ability for printers to produce parts for themselves, other or new printers enables a new level of open-source hardware development, where a part can be designed, fabricated and tested with minimal time and capital investment. The new design can then be shared with the pool of users for it to be implemented across any pre-existing devices. While this has been useful during the development of the printer in this thesis, even more significant is the possibility of a wide audience getting the ability to replicate the result and contribute.

Open design and self-replication have uniquely positioned 3D printing in a set of circumstances in which citizen science constitutes a significant contribution to the overall development in the field, which has lead to the explosive growth of the technology. I incorporate the concepts of open design and self-replication in my method, as they are crucial to gaining traction with the community.

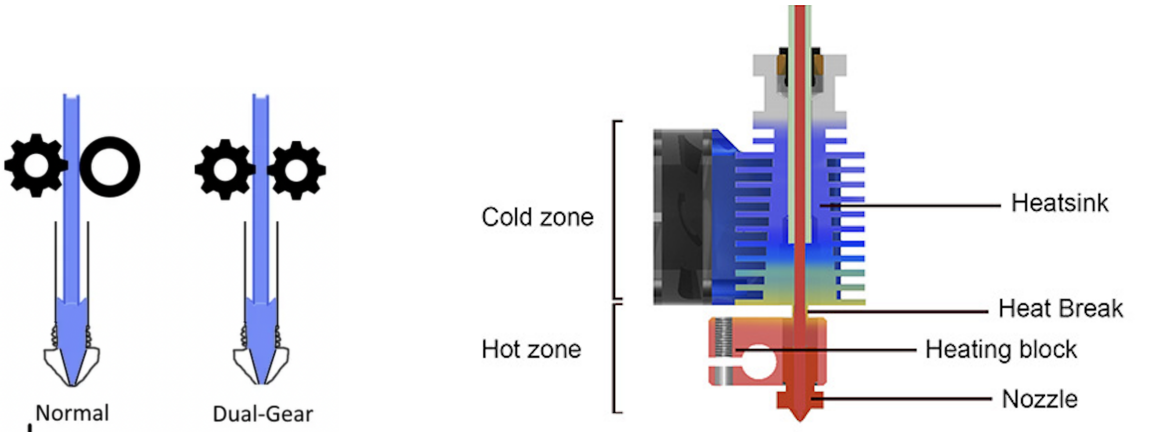
1.1.2.2 Working principles

A FDM printer consists of four main mechanical sub-systems - extruder, hot-end, build-plate, and kinematic system. The extruder feeds material to the hot-end, by mechanically pushing the filament. In the hot-end, the material melts and gets pushed out of an orifice onto the build-plate. Meanwhile the kinematic system is moving the hot-end relative to the build-plate, such that molten material coming out of its orifice gets deposited along a path. These three systems work in unison to produce a print.

The extruder is driven by a stepper motor for precise control over the extruded amount of filament. It acts on the filament mechanically, using a fine-tooth gear that grips onto the filament,[4] and pushes it forward as the gear rotates, as shown in Figure 1.9a.[5] The extruder can be mounted directly above the hot-end in a configuration called direct extrusion. Alternatively, it can be mounted elsewhere with the filament being pushed through a Bowden tube, which would be referred to as Bowden extrusion. Both configurations are shown in Figure 1.10. Bowden extrusion has the

[4]Some systems have an extruder gear on both sides of the filament, for improved grip, and the extent of grip can be adjusted by the user using a tensioner.

[5]Other systems also exist, such as screw extruders for pelletized material.



(a) Schematic representation of extruder setups[15]

(b) Schematic representation of hot-end, showing components and temperature[16]

Figure 1.9: Extruder and hot-end systems of a FDM printer

drawback of having slack in the filament, as it slightly contracts and expands as it is being pushed through the full length of the tube. This makes it particularly difficult to print flexible materials using a Bowden extruder. However, it has the benefit of reducing gantry mass, which allows for a greater print speed compared to direct extrusion.

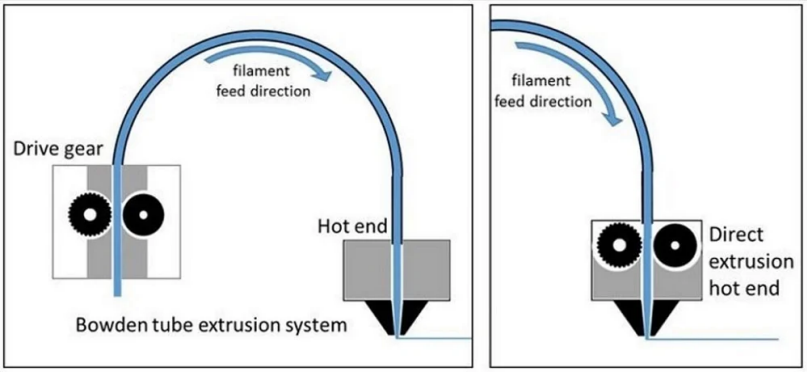


Figure 1.10: Schematic comparison between Bowden and direct extruders[17]

The hot-end can be split into the melt zone and cold zone, as shown in Figure 1.9b. The melt zone consists of the nozzle and heat-block. The cold zone consists of the heat-sink and fans. Both are connected by the heat-break. Filament being pushed into the hot-end passes through the cold zone into the hot-zone. The heat-block is at the melting temperature of the material, which is usually accomplished by a heating cartridge mounted in the heat-block. The temperature is controlled using a Proportional-Integral-Derivative (PID) controller, which uses a thermistor, also mounted in the heat-block, as feedback. As additional filament is being pushed into the heatblock, the back-pressure pushes molten material through the orifice of the nozzle. As material exits the nozzle, it solidifies quickly, as it cools down due to convection, usually accelerated by a part-cooling fan. The heat-break limits heat transfer from the melt zone to the cold zone, by having a small cross-section limiting thermal conduction, and sometimes also being made of a material less heat conductive. Any heat that does creep up to the cold zone is immediately dissipated by the heat-sink, which is cooled by a fan. If heat is allowed to creep up the cold zone, too much filament melts, clogging

the hot-end assembly.

The build-plate is the surface onto which the part will be built. It is typically heated to aid with adhesion between the substrate and part, also regulated by a PID controller. The build-plate typically consists of a sheet that heats up, and another changeable sheet.

The kinematic system is the mechanism that facilitates the movement of the End Effector (EE) - the hot-end assembly - relative to the build-plate. There are many different kinds of kinematic systems used for FDM, which are detailed in Section 2.1.1. For typical FDM the kinematic system has 3 Degrees of Freedom (DoF) - X, Y and Z. As material is deposited through the hot-end onto the build-plate, the speed and distance at which the two are traveling relative to one another determine the height and width of the track.

Figure 1.11: 3D printing software stack

To go from 3D designed part to 3D printed part there is a series of software involved, which constitutes the software stack - CAD, slicer, firmware. The Computer Aided Design (CAD) software is the environment in which engineering parts are designed. The model is exported to a file type which is supported by the next software in the stack - the slicer.[6]

A slicer is a software which turns the CAD model into toolpaths. These are the paths that the hot-end will follow to deposit material in the correct places to build the part. It is aptly referred to as a slicer because it slices parts into layers to be printed. Each layer is a cross-section of the part. Depending on the needs various part infill strategies can be applied. Printing just the perimeter (also known as vase mode, spiral mode, or 0% infill) produces a hollow shell. Whereas printing solid (also known as 100% infill) produces a quasi-solid part, as FDM part infills are inherently porous due to small gaps between tracks. Usually a partial infill is used, to get strength and stiffness while maintaining light weight and reasonable printing time. Toolpaths are exported as Geometric code (G-code) - a Computer Numerical Controlled (CNC) programming language. G-code uses a simple syntax, in which each line of code commands the printer perform certain movements, given in coordinates, at a certain speed. By stringing together a series of these commands the entire toolpath is defined. This process is detailed in Section 4.1.1.

The firmware is the software that runs on the printer's board. It reads G-code and commands the printer's hardware to execute the G-code commands. This includes actuating the motors of all axes to follow the defined toolpath, and running the PID loops for hot-end and print-bed heating.

[6]The .stl format is commonly used as it is widely supported. However, it introduces additional error due to the meshing process, in which the CAD model is converted into a tesselated mesh consisting of triangles. In 2022, the popular PrusaSlicer started supporting .step files, which retain the original geometry of the part, and is widely used in CAD data exchange.

1.1.2.3 Challenges in FDM

Based on the working principles stated above, there are certain challenges associated with the FDM process, some of which are common among many groups of AM technologies.

Overhanging geometry Since material is deposited through a nozzle that is normal to the build-plate, it is difficult to print overhanging geometry. The track which is being printed can't be printed mid-air, and should be partially supported, as shown in Figure 1.12. It is typically considered that FDM processes can maintain quality at an overhang angle of up to 45 degrees, beyond which the quality of the surface will suffer, or the print will fail, as shown in Figure 1.13. Straight overhanging geometry can be printed between two structures, called bridges, as it is supported on either side. However, longer bridges are more difficult to print. Additionally, cooling capacity or lower temperatures increase overhang capabilities. If the material solidifies fast enough as it is being deposited, it can better support itself.

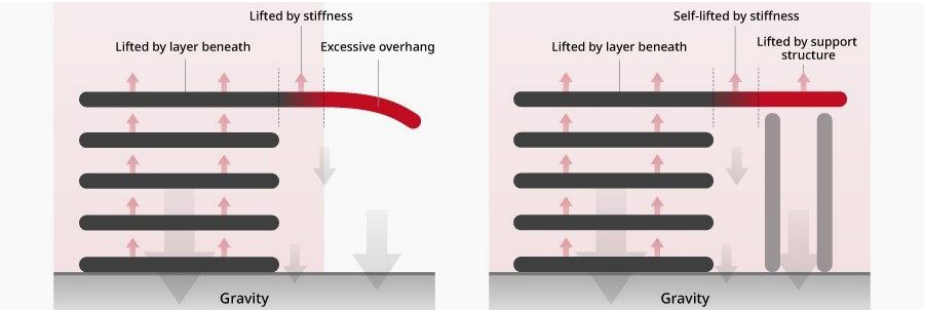


Figure 1.12: Principle of using support structures as scaffolding for overhanging geometry[18]

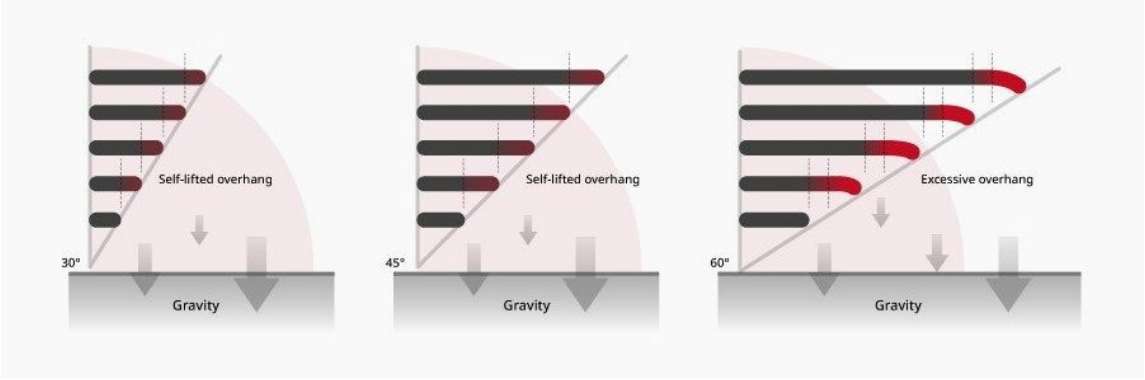
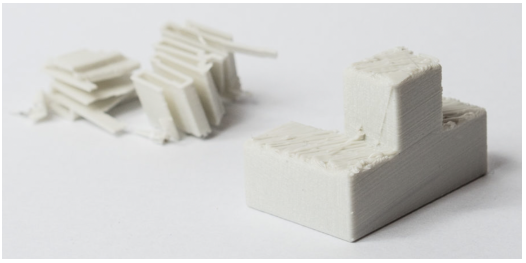


Figure 1.13: Typical overhang angle[18]

When overhanging geometry is required, support structures are often used. Support structures are a sacrificial part of the print, the purpose of which is to act as scaffolding for overhanging geometry to be printed on top of. These structures are removed as a post-processing step. The difficulty of removal can vary wildly depending on how well the structures have been designed, and how intricate and delicate the supported geometry is. Though they can usually be removed without tools, or with just pliers, it still adds additional labor to the full printing process. Traces of support structures often leave a rougher surface finish, as shown in Figure 1.14a



(a) Damage to supported surface after removal of support structures

(b) Typical support structure, tree or organic support structure

Figure 1.14: Surface damage from support structures, and organic support structures[19]

To reduce the manual labor associated with support removal, and improve the surface quality where they have been present, a soluble support material can be used. By printing the support structures out of e.g. PVA, the finished print can be bathed to dissolve the supports instead of requiring manual removal. For this a multi-material printer is required, which uses either a dual hot-end, hot-end changer, or material changer. Such systems have a higher upfront cost, and also produce significant material waste, as material has to be purged between changes (the extent depends on the specific system). The variable costs also significantly increase, as soluble material filament is several times more expensive than the typical materials that are printed with, and require additional care to be kept dry as they are hygroscopic.[7] Recently, organic or tree supports have become available as an alternative, reducing the contact area between support and part, as shown in Figure 1.14b.

Accuracy Geometric inaccuracies in printed parts can occur in many of the steps of the generic AM process, as described by Gibson et al.[2] The critical sources of inaccuracy for FDM are those that stems from the deposition of the material - extrusion. The layered deposition combines with the arc-shaped sides of the track profile results in a relatively high surface roughness.[20] As shown in Figure 1.15, there are voids between the deposited oblique surface and the theoretical flat surface being printed, marked e. Lower the surface angle α and higher layer height t increase e.

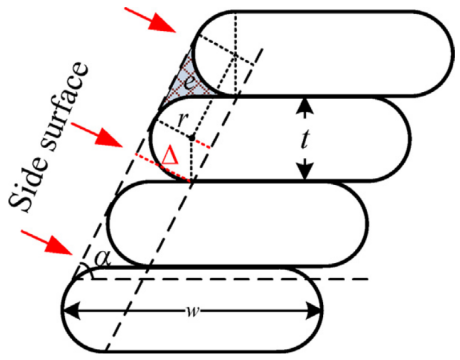


Figure 1.15: Cross-sectional view of deposited tracks, showing inaccuracies in oblique surface manifesting as poor surface quality[20]

[7] To reduce the amount of water-soluble material used, the bulk of the support structures can be printed in the part material, with only the interface layer being printed in soluble material.

Anisotropy AM parts exhibit mechanical anisotropy, and it is the most prevalent in FDM due to the insufficient interlayer bonding.[21] As a track is being deposited, tracks on the previous layer have cooled to an extent that limits the entanglement of polymer chains between the two. Thus FDM parts are generally weak in the build direction, i.e. the direction normal to the build-plate, as shown in Figure 1.16. This limitation requires careful consideration of build direction for functional parts.[22]

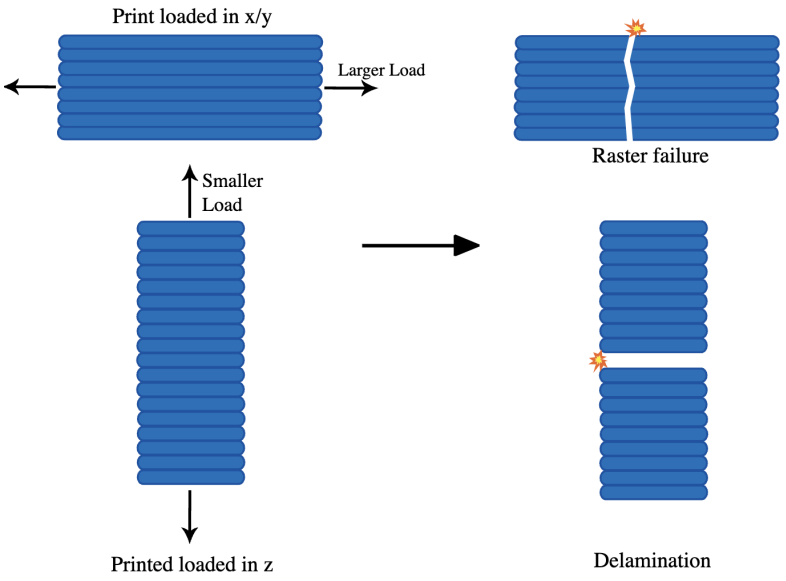


Figure 1.16: Part strength and failure mode depending on loading direction[22]

1.1.3 Multi-axis AM

Up until now we have discussed planar FDM with 3 axes, as it was conceptualized in its inception. However, there is no inherent limitation to the DoF or axes that a FDM printer can have. Multi-axis FDM incorporates the rotation of the printhead relative to the build-plate, as shown in Figure 1.17. [24][25] With this non-planar approach starting to emerge, it shows potential in addressing all of the challenges just discussed - overhangs, anisotropy, and surface quality.

1.1.3.1 Benefits

Improved surface quality The poor surface roughness on FDM prints stems from what is referred to as the stair-step effect. Due to the layer-by-layer approach of typical FDM printing, the print's Z-direction resolution can only be as low as the layer height.[2] This is particularly noticeable for surfaces that have a slight incline or curvature. While some of the negative effects can be diminished by using a small layer height, it significantly increases the print time and weakens the part.[26] Instead, multi-axis AM can reduce the stair-step effect with no penalty to print speed or part strength. There are different multi-axis AM strategies that reduce the stair-step effect. E.g. curving the top layer eliminates the stair-step effect entirely,[27] resulting in a smooth surface as shown in Figure 1.18b.

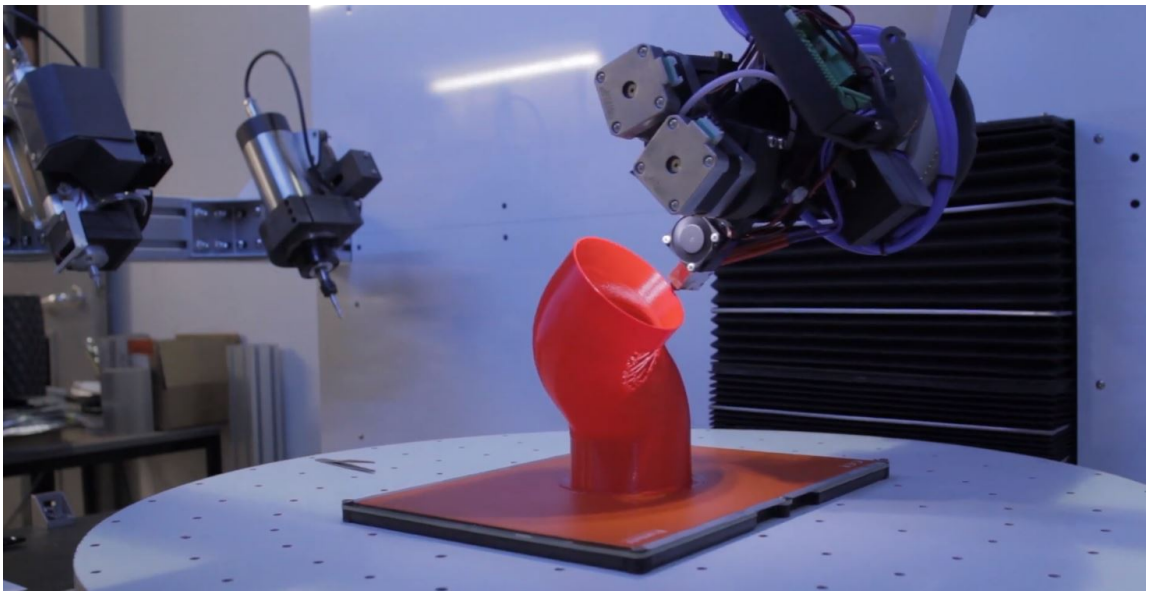
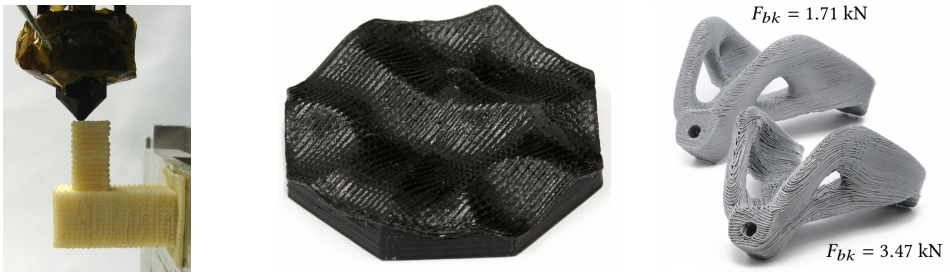


Figure 1.17: 5-axis FDM printer by Q5D technologies[23]

Reduced waste One of the rudimentary benefits of multi-axis AM is the ability to print significant overhangs without using any support material. Multi-directional slicing involves changing the print direction, such that a low overhang angle can be enforced, even if the overhang angle relative to the build-plate is high,[28] as shown in Figure 1.18a. Support-free printing is also the general reason why DED processes require a multi-axis machine in most cases, as support structures are typically not used due to how impractical it would be (especially for WAAM). This has caused multi-axis process planning to mature much faster for the DED process, as it is much more of a necessity, while overhangs for FDM parts can always just be solved using water-soluble supports. Nevertheless, obsolescence of support structures for the FDM process is still beneficial, as it would reduce the total print volume, often by several times, reducing print time and cost.[29] Additionally, time and cost related to the post-processing step of support structure removal, a major cost driver due to its labor intensity,[30] is effectively eliminated.



(a) Large overhang printed support-free[31] (b) Smooth, curved top layer[17] (c) Controlled anisotropy improving part strength[32]

Figure 1.18: Benefits of multi-axis printing

Controlled anisotropy Parts printed using the FDM process exhibit strong anisotropy, as explained in Section 1.1.2.3. Thus parts printed using the FDM process are significantly weaker

along the print axis, and loading them on that axis will result in failure by delamination. Utilizing multi-axis AM part anisotropy can be controlled, by selectively choosing different build directions for sections of a part, empowering engineers to design FDM parts that aren't necessarily weak in the build direction.[32]

1.1.3.2 Challenges

The process also comes with its share of challenges, both in the realm of mechanical and software design. The hardware of a multi-axis 3D printer is more elaborate, as additional axes of movements are required, making the kinematics more complex and being more prone to inaccuracy. The software for path-planning must be significantly more elaborate, as complex algorithms are required to effectively utilize the additional degrees of freedom to execute on the aforementioned benefits.[33][34]

Mechanical design The addition of additional axes makes the mechanical design more elaborate. Additional joints and links, alongside motors to drive them, increase the complexity of the design. Furthermore, adding elements to the kinematic chain always demands careful attention so as to not introduce additional kinematic error.

Path-planning Typical slicers for FDM will follow the previously described heuristic, of cutting the part into cross-section that are parallel to the build-plate, at a certain spacing - the layer height. Each of these 2-dimensional cross-sections are printed with perimeter and infill, subsequently on top of one-another, stacking them into a 3-dimensional volume. To attain the benefits associated with multi-axis FDM, parts must be sliced differently, requiring new algorithms or the augmentation of existing ones. Additionally, collision prediction and/or avoidance must be implemented, as revisiting previous layers and adding rotational movement to the nozzle can lead to interference between print-head and workpiece or print-bed.

Firmware Additive and subtractive manufacturing machines generally use the same principles of G-code for operation. In both cases a tool - whether it be the spindle in the case of milling, or the hot-end in the case of FDM - is moved along a toolpath defined in the G-code. The differences are in some nuances between the specific hardware of the machines. One of the critical differences is that FDM is typically 3-axis, and thus all out-of-the-box firmware for FDM are designed for 3 axes. Multi-axis FDM increases the complexity of the firmware, as it has to translate the additional A and B values into movements for their respective motors. With multi-axis CNC machines, Inverse Kinematics (IK) typically needs to be implemented in the axis control, as movement of the rotational axes introduces offsets to the linear axes.

1.2 MOTIVATION

Multi-axis FDM is an emerging direction in the field of AM. Despite the clear advantages that address the fundamental limitation of current FDM printers, the technology is still not widely adopted. Hardware and software for multi-axis are not widely available. In fact, there are no

multi-axis FDM printers that are as accessible as conventional desktop FDM printers. What lead to the democritization of FDM technology was the hobbyist community that sprung up around the open-source RepRap project. The crowdsourcing of contributions accelerated development. And with self-replication new hardware developments could be pushed to end-users in a fashion before unseen.

By showing the utility of the multi-axis approach, and demonstrating how it can be accessibly implemented, enough interest within the 3D printing community could be garnered to spur widespread involvement, which would also accelerate overall development in AM.

1.3 RESEARCH OBJECTIVE

The objective of this research is to investigate whether an accessible multi-axis FDM printer can address the shortcomings of conventional FDM without introducing new drawbacks. By showcasing the advantages and feasibility of the multi-axis approach and its implementation, the research intends to generate sufficient interest within the community to promote widespread involvement and accelerate the overall development of AM technology.

This is achieved by designing and building an open-design multi-axis FDM printer on an accessible platform, and using it to print sample parts sliced with an accessible path-planning approach.

Chapter 2

State of the art

THIS CHAPTER WILL EXPLORE the state of the art in the broader fields of non-planar and multi-axis FDM. Firstly, **printer hardware** will be considered, to examine what kind of multi-axis printers have already been developed - to what extent those have been successful and how to improve upon them. From studying the properties of past multi-axis printer designs critical design requirements can be extracted, which will guide me in the design of my own printer.

To characterize the quality of a 3D printer, a benchmark process should be employed. Thus, subsequently, existing **benchmarking** procedures for 3D printers will be examined, to determine the most appropriate for this work. There is no universal benchmarking procedure, so various options found in the literature will be considered and evaluated based on their applicability for specifically Multi-Axis FDM.

Finally, **path-planning** strategies and algorithms will be examined to find the most appropriate approach to slice the benchmark part for printing on a multi-axis FDM printer. Unlike for conventional FDM, there is no universal multi-axis slicing strategy. Instead a variety will be considered to find the most appropriate for the chosen benchmark parts.

By reviewing the state of the art in printer hardware, benchmark part selection, and path-planning, I lay the foundation for exploring multi-axis FDM. This chapter will provide the necessary insights and knowledge required to design and develop my own variant of a multi-axis FDM printer and generate the toolpath G-code for printing benchmark parts.

2.1 PRINTER HARDWARE

2.1.1 Kinematic structures

At the core of a 3D printer's hardware is its kinematic structure - the interconnection of links and joints of the mechanism.[35] This structure facilitates the spatial movement of the so-called hot-end, which is the system that melts material and subsequently deposits it. In tandem, these two systems execute the fundamental function of a FDM printer - the hot-end deposits material, and the kinematic system moves the hot-end relative to the build-plate such that material is deposited where needed. Choosing a suitable kinematic system is important to the objective of the work for two reasons. Firstly, the printer should perform well in terms of accuracy, and kinematic error is a key source of potential inaccuracy in the 3D printing process, as explained in Section 1.1.3.2. Secondly, the intention of making the printer design accessible, as the cost or complexity

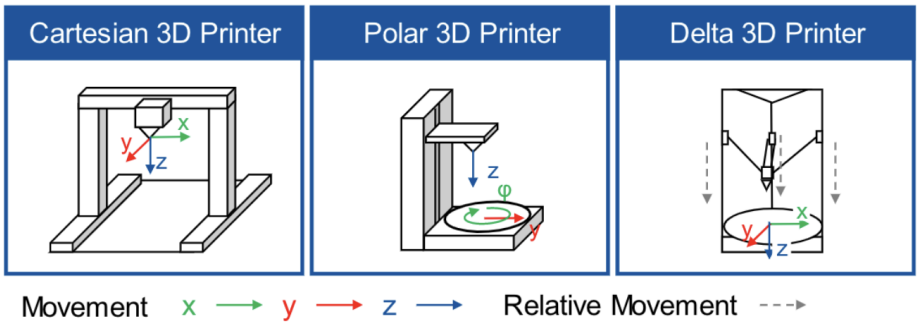


Figure 2.1: Kinematic structures for conventional FDM printers[37]

of different kinematic systems influence the choice. A wide variety of kinematic structures have been shown on conventional FDM printers, such as the cartesian, delta and polar, shown in Figure 2.1. The broader category of cartesian kinematic structures is by far the most common, and has can be further split into 40 different kinematic structures based on which axes actuate the bed vs. the head, which are stationary vs. moving, and which are cantilevered vs. supported on both ends,[36] some of which are shown in Figure 2.2. It is referred to as the cartesian kinematic system, because the directions of the 3 DoF relate to the X, Y and Z axes of the cartesian coordinate system.

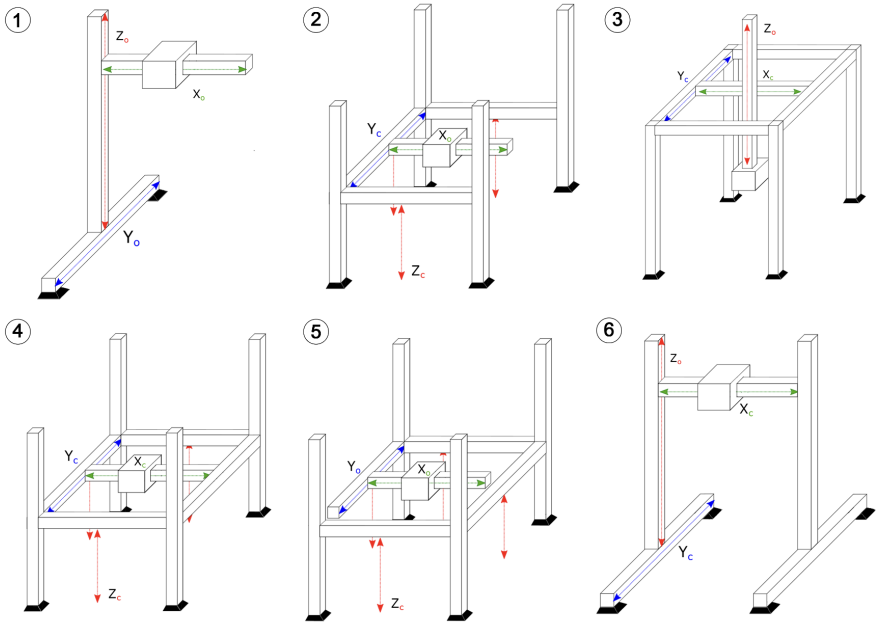


Figure 2.2: Variations of cartesian kinematic systems[36]

The most recent development in kinematic systems for conventional FDM printers is the CoreXY variant of the cartesian design, which boosts printing speed by reducing the gantry mass.[38] The X and Y motion is driven using a pulley system that allows for the driving motors to be stationary.[1][39] The resulting decrease in moment of inertia for the gantry significantly improves movement speeds and reduces vibrations, which in turn enables higher print speeds and better quality.

[1]The hot-end must actuate on the X and Y axes, whereas the bed is actuated along the Z axis.

Alternatively, the reduction of gantry mass can be used to offset the increased mass of e.g. a multi-material system. The CoreXY motion system is used in the highly popular Bambulab X1 desktop FDM printer, named by Time Magazine as one of the best inventions of 2022 in the Robotics category[40].

All of the mentioned kinematic systems have 3 DoF - 3 linear and no rotational - which is the minimum required DoF to perform the typical FDM workflow of depositing material on the XY plane, then moving it to a new Z height to deposit the next layer. There are various kinematic systems with additional DoF, to enable what is referred to as multi-axis AM. In the following sections various different multi-axis kinematic structures will be examined, and evaluated on their utility and applicability for non-planar AM.

2.1.2 Serial vs. parallel manipulators

Manipulators can be categorized into two groups - serial and parallel. This refers to whether all the linkages of the kinematic structure are connected in a series and motion is transmitted sequentially, or whether the linkages are distributed and the motion is shared, as shown in Figure 2.3. Assuming the same input error, parallel mechanisms are inherently more accurate than serial mechanisms, as the input errors are averaged as opposed to cumulative.[41] However, that is merely a theoretical judgement, and in practice, there are many more factors beyond kinematic error that will influence the accuracy of prints. On several occasions cartesian printers have been shown to outperform their delta counterparts, despite the supposed superior accuracy of the delta kinematics.[42][43] Manipulators of the two groups also differ in load capacity and reach.[44][45]

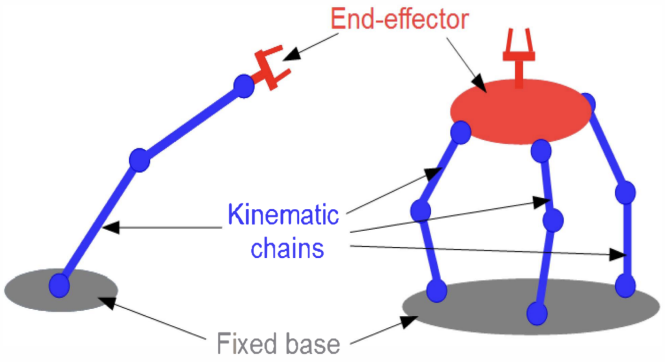


Figure 2.3: Serial and parallel manipulators[45]

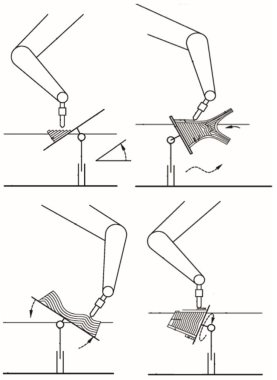
2.1.3 Robotic arm

The robotic arm is the quintessential serial manipulator. Typically 6 DoF, these manipulators cover the full 3 translational and 3 rotational DoF's. In combination with the high reach, robotic arms are versatile and useful for a wide variety of automation tasks, however, at a steep cost.

Robotic arms are commonly used to illustrate multi-axis path-planning. Opting for a robotic arm eliminates the need to develop any hardware solution, instead making use of an commercial off-the-shelf (COTS) solution just to execute the software. Different kinds of robotic arm setups have

(a) Hot-end on end effector[46]

(b) Build-plate on end effector[47]



(c) Combination of robotic arm with other platform[24]

Figure 2.4: Robotic arm configurations for FDM

been used for multi-axis AM; hot-end as the EE, printbed as the EE, or a combination of both, as shown in Figure 2.4.[25] This is the prevalent form of kinematic structure for publications about multi-axis AM, because the focus of those publications concern path-planning.[47][46][28] The wide range of motion and high reach gives software researchers the most freedom. However, these systems are too expensive to be adopted in the consumer or prosumer AM market.

2.1.4 Combined 3+2 axis mechanisms

Another way of achieving multi-axis movement is by combining mechanisms, as is commonly done for CNC machines. E.g. for a 3D printer, instead of using a 5-axis manipulator, a 2-axis manipulator could be added to an existing 3-axis printer, effectively augmenting it to 5-axis motion. These combined mechanisms can be categorized into 3 groups, based on which of the two additional axes are actuated at the head or table - head-head (HH), table-table (TT), and head-table - as shown in Figure 2.5. The reason why only 5 DoF are required for FDM is that the output of the nozzle is circular. Rotating around the circular nozzle would have no effect on the outcome, rendering the final 6th rotational DoF redundant.[2]

2.1.4.1 Table-table configuration

Micinski et al show a parallel manipulator actuating the printbed of a Prusa MK3 printer, shown in Figure 2.6a, stating an "attempt (...) to build a simple and low-cost device".[49] That particular design illustrates a failure to understand critical design requirements, as the resulting mechanism is a system of complex linkages driven by an excessive amount of motors. No particular rationale is given for the reasoning behind the complex design. While a non-planar print is produced using the printer, it does not demonstrate impressive capabilities, as the chosen geometry would not be challenging to print on a conventional printer.

[2]This would not be the case if a non-circular nozzle is used, as has been shown in some concepts.

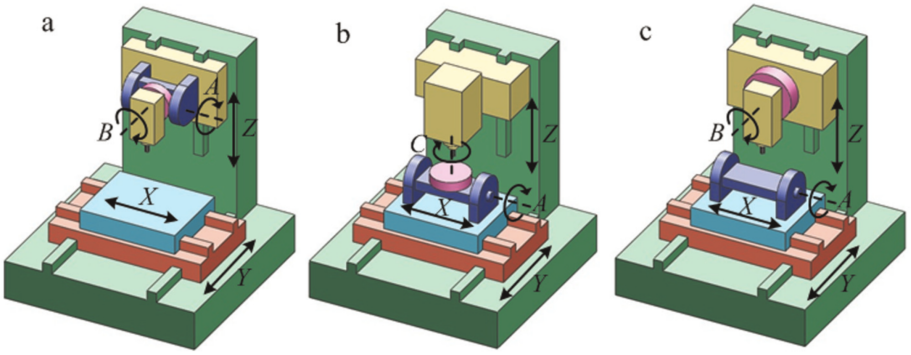
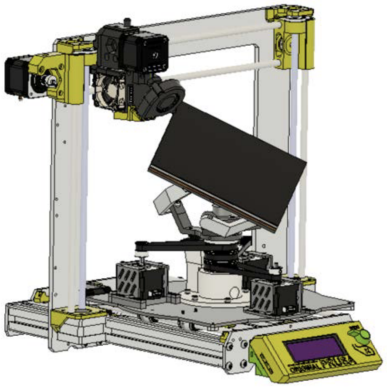
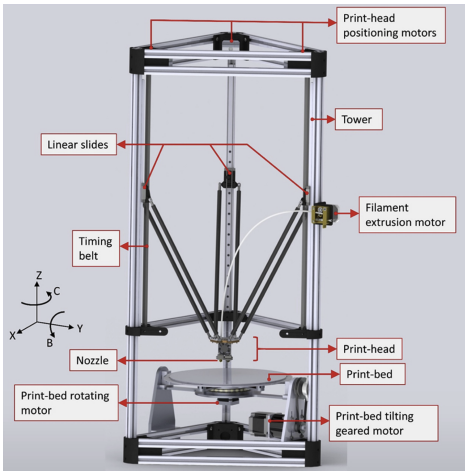


Figure 2.5: Kinematics configurations for 3+2 axis mechanism: (a) head-head; (b) table-table; and (c) head-table [48]

Isa and Lazoglu implement a build-plate that rotates and pivots, as shown in Figure 2.6b, to enable the additional 2 axes.[50] For the base printer they chose delta kinematics. Three reasons are given for choosing the TT configuration over the HH: (a) simplicity of the design; (b) concern of adding load to the linkage arms of the head; and (c) "keeping the nozzle aligned with gravitational acceleration assists in material deposition". The design is shown to print complex parts that illustrate some benefits of multi-axis AM, however defects are present in the parts, explained by the authors as a result of kinematic errors.[50]



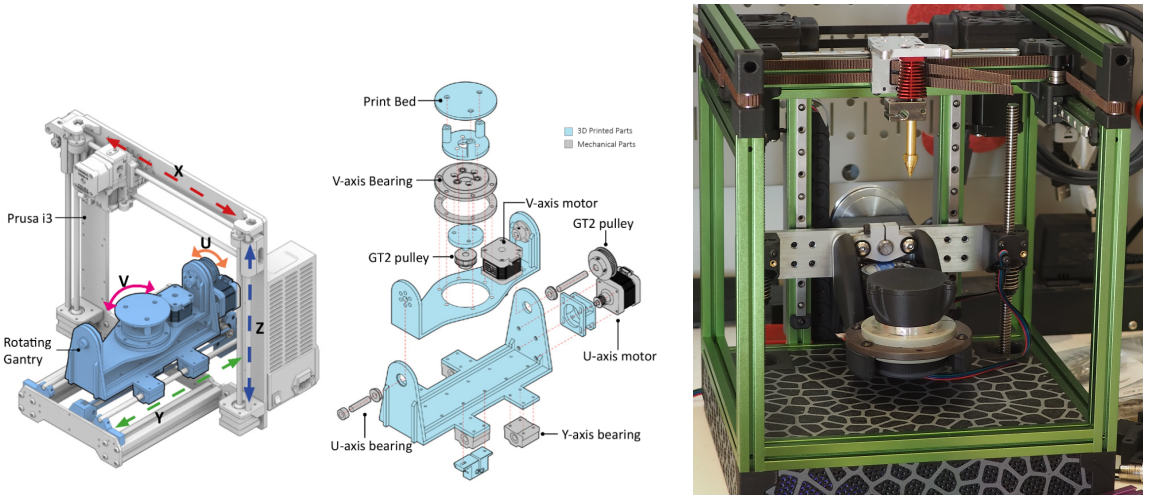
(a) Parallel manipulator for Prusa MK3[49]



(b) Rotary table on delta printer[50]

Figure 2.6: Table-table configuration printers

In 2022 Hong et al published Open5x, an open-source 5-axis printer design and conformal slicing algorithm.[51] The printer design is also a rotary pivoting build-plate, but it is designed for the popular Prusa i3 MK3 printer, shown in Figure 2.7a. Additionally, most parts can be 3D printed, making it self-replicating. The Prusa MK3's build-plate is normally mounted on the Y-axis gantry (in the so-called "bed-slinger" configuration), so with this setup the entire rotary table is moved, which is a significant mass. This incurs a significant speed and accuracy penalty on the Y-axis, the high moment of inertia translates to lower axis velocity and more vibration. Additionally The printed parts for the rotary table are large, coming close to typical build volumes of desktop 3D



(a) Original Open5x rotary table for Prusa MK3[51]

(b) Open5x variant for Voron[52]

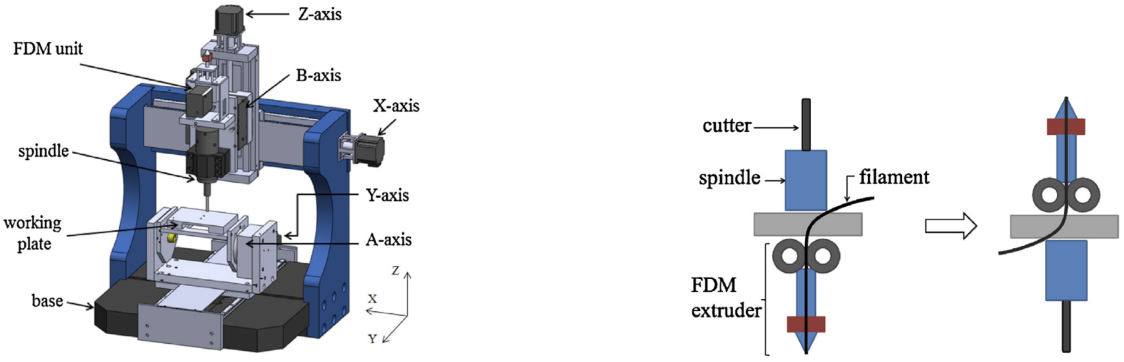
Figure 2.7: Open5x printers

printers,[3] making self-replication difficult. The design has been shown to produce complex, albeit small, prints. An example of an improved Open5x design is shown by a maker under the alias BrendonBuilds, in Figure 2.7b. In his take on the rotary table, it is mounted on the Z-axis of an XY-gantry style printer. A key benefit of this design change is that, unlike for the original design, the rotary table requires less movement, as it is disconnected from the most active X and Y axes. This design also shows a significant improvement in clearance, best illustrated by a sample print in which a tube is printed with extreme overhangs and crossing the plane of the build-plate.[52] It does, however, represent a departure from the idea of accessibility, as it is built on a premium 3D printer plaform instead of the common Prusa. Using a combined 3+2 axis solution has the benefit of being cheaper than a 5-axis manipulator, not just in total cost, but also considering that existing 3-axis machines can be retrofit.

2.1.4.2 Table-head configuration

Lee et al developed a 5-axis table-head FDM printer, that cleverly uses the extra axis on the head to also enable machining operations on the same device.[31] Such a process can be referred to as a hybrid additive-subtractive manufacturing process, or just hybrid AM. This combination of two opposing manufacturing approaches can yield the benefits of both processes while counteracting the technological limitations of each.[53] One particular use of a hybrid AM process is the improvement of surface quality, by machining a finishing pass to eliminate the aliasing or other surface deviations, common in AM.[54] Hybrid AM processes are rarely ever used for FDM, due to the significant increase in cost and complexity, and the relatively limited benefit, compared to e.g. Wire Arc Additive Manufacturing (WAAM) which produces a significantly poorer surface. In the design, shown in Figure 2.8a, the B axis also acts as a tool-changer between the spindle, used for machining, and the extruder. As shown in Figure 2.8b, the spindle and extruder are mounted on opposite ends of the B axis, such that the two tools can be switched without any additional moving parts.

[3] Usually around 200x200mm build area



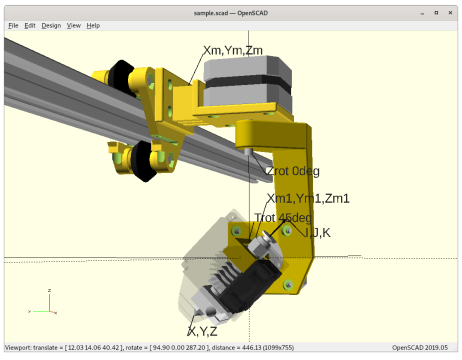
(a) Kinematics of the machine

(b) Tools mounted on the B-axis

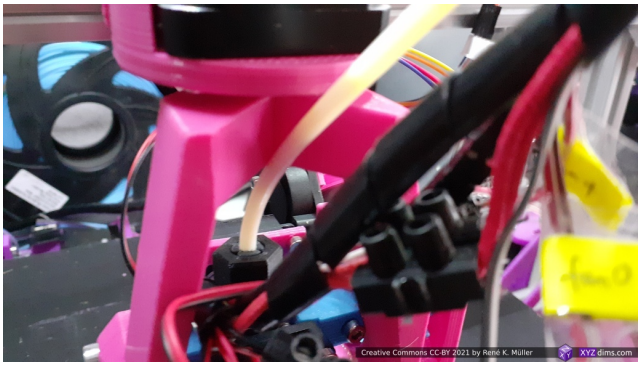
Figure 2.8: Hybrid AM-SM 5-axis machine[31]

2.1.4.3 Head-head configuration

A head-head solution is developed by Muller, shown in Figure 2.9a.[55] In the design the additional axes actuate the hot-end to produce a rotary and pivoting motion. Various drivetrain solutions are shown, including the use of a worm gear to increase torque of the pivoting axis. The retrofit hot-end is shown on a typical cartesian printer of the bed-slinger variety, meaning that the assembly moves with the X and Z axes. Since the build-plate is unaltered, this approach has significantly increased build area and volume compared to the previously considered TT and table-head configurations.



(a) Model of printer



(b) Bowden tube interference with hot-end and gantry geometry

Figure 2.9: Pentaxis (PAX) printer[55]

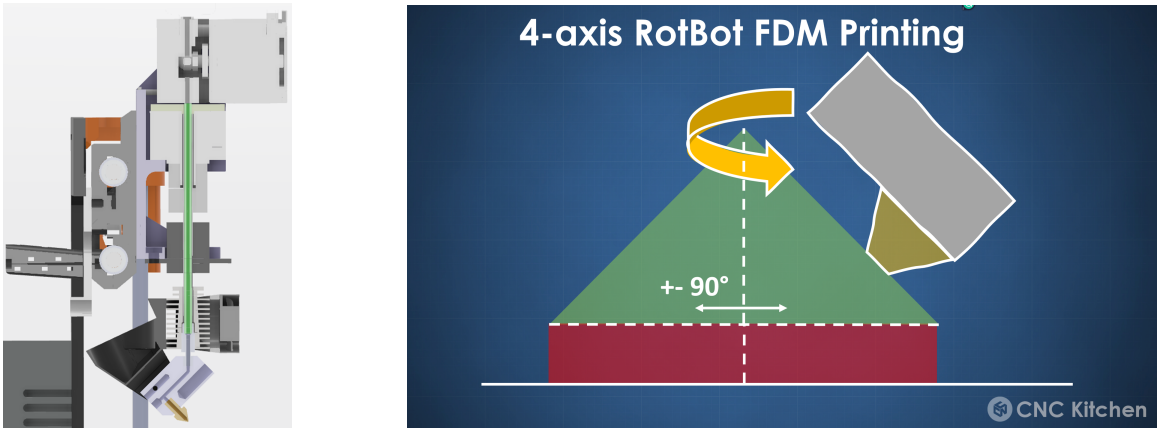
One issue highlighted with this approach is the routing of the filament or bowden tube. Filament material is fed to the hot-end either directly in the case of a direct extrusion system, or through a bowden tube. Either way, this can limit the range of motion of the hot-end as the stiff filament or bowden tube interfere with the gantry as shown in Figure 2.9b. This also renders continuous rotations of the rotary axis impossible unless a slip joint is used both for the filament and electrical wires going to the hot-end.[4]

[4]This has been implemented on the RotBot, explained in Section 2.1.5

2.1.5 Other kinematics configurations

4-axis conical printer

A specific 4-axis printer design, the RotBot, developed by Wütrich et al, aims to deliver support-free printing with less complexity than system with more axes.[56] In this printer, the hot-end, which is mounted at a 45 degree angle from the Z-axis, rotates around the Z axis, as shown in Figure 2.10a. The 45 degree angular offset increases the envelope at which it can print overhangs to around 90 degrees; meanwhile the Z axis rotation ensures that this extreme overhang printing can be done in any direction, as shown in Figure 2.10b. Using a particular slicing strategy - conical slicing, explained in more detail in Section 2.3.3.2 - this printer can print certain parts with significantly higher overhangs than conventional printers can. In order to facilitate continuous rotations it uses a slip ring for the electrical connections of the hot-end, and a special slip connection for the filament.



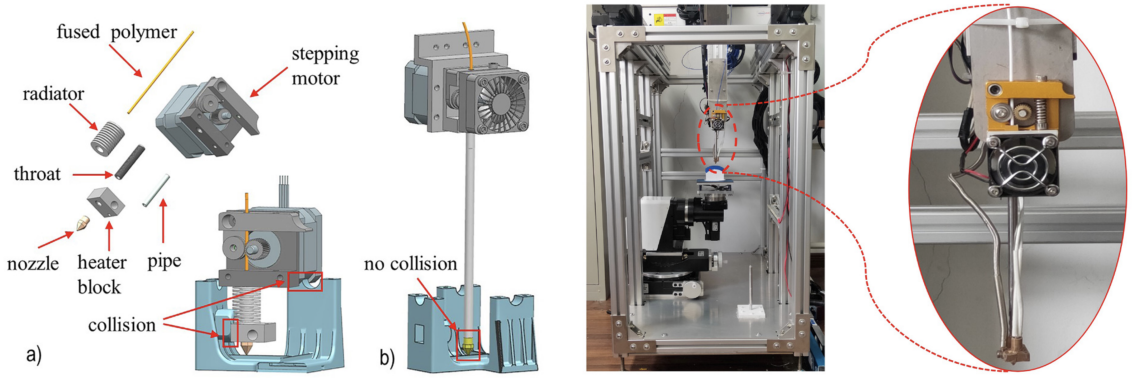
(a) Section view of print-head

(b) Overhang printing envelope of RotBot

Figure 2.10: RotBot 4-axis printer[56][57]

2.1.6 hot-end hardware

Not only kinematics systems play a role in hardware design for multi-axis AM. A long-throated extruder proposed by Liu et al may solve some of the interference issues common in non-planar printing. Conventional hot-ends are busy around the nozzle orifice with ducts for part-cooling and often a sensor for bed leveling. Naturally this is not an issue in conventional printing, as the flat layer approach ensures no part geometry is ever higher than the nozzle's tip. Figure 2.11a illustrates how interference can be an issue in non-planar printing, and how a long-throated nozzle can solve that. However, the imagined long throated nozzle concept would not work, as material needs to be melted just before the nozzle, and no provisions for that have been made. In Figure 2.11b a heat block has been added before the nozzle, negating most of the clearance. Cooling is also lacking in this setup - no part cooling is present, which reduces print speed and quality. Additionally, the cool zone (cooled by the hot-end fan) is very far from the nozzle. This means that the entire volume inside the long throated nozzle acts as a hot zone, which could increase clogging. The extended nozzle by nonplanar.xyz, shown in Figure 2.12a, offers a comfortable



(a) Concept for long throat to reduce interference (b) Implementation with heat block

Figure 2.11: Long throated nozzle by Liu et al[59]

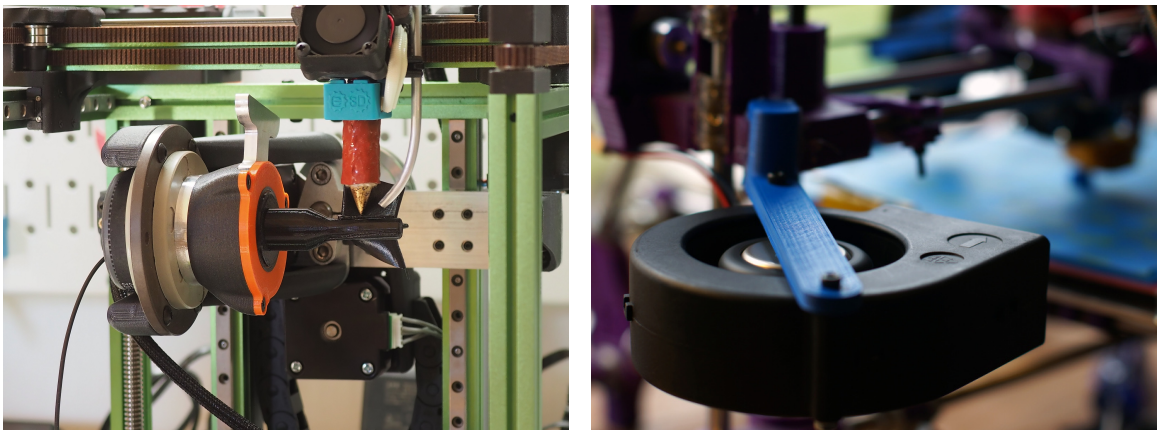
middle ground.[58] The nozzle is extended by about 20 mm, significantly improving clearance, as the heatblock also remains further from the nozzle. To maintain the heat, the nozzle is fitted with a heat-insulating sleeve.

A different approach for a slim hot-end is the open-source Goliath hot-end.[60] Instead of using a bulky heat block with a heating cartridge for material melting, it uses a slim tube wrapped with resistive Nichrome wire, as shown in Figure 2.12b. While this particular hot-end concept was not expressly designed with the intention of improving clearance for non-planar printing, it is a great option for it.

(a) Extended nozzle with heat insulator by non-planar.xyz[58] (b) Goliath hot-end with slim heat block[60]

Figure 2.12: Variants for improving clearance around nozzle tip

None of these solutions address part-cooling, which, while not strictly necessary, improves printing speed and quality. In the Open5x Voron variant the part cooling fan is installed further away, and airflow is directed towards the nozzle tip using a tube, as shown in Figure 2.13a. This pro-



(a) Remote fan with slim tube/duct to guide airflow (b) Whole build-plate cooling[61]

Figure 2.13: Part/layer cooling solutions for multi-axis

vides adequate part cooling while taking up minimal space around the nozzle. Alternatively, fans mounted to inside the build volume can cool the general area of the nozzle tip, as shown in Figure 2.13b. Though inefficient, this method is popular for high-speed printing, as it reduces the gantry mass. For multi-axis printing it could be valuable thanks to the total elimination of one cooling system on the hot-end aiding with clearance.

2.2 BENCHMARK PART SELECTION

2.2.1 Benchmark parts

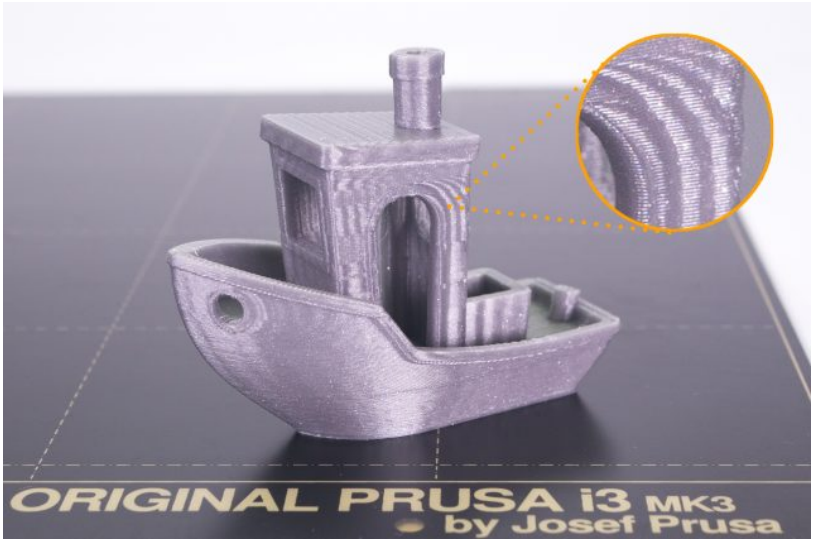
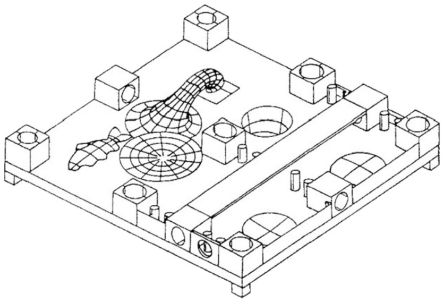


Figure 2.14: Quintessential FDM benchmark part - Benchy[62]

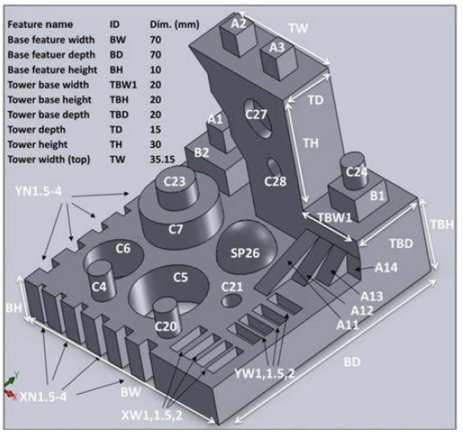
The performance of printers is evaluated by printing benchmarks. The benchmark part contains a variety of geometry in different sizes, and particularly such that is difficult to print. This informs

you about the printers performance not just in dimensional accuracy of printed parts, but also in a plethora of other factors, such as part cooling, bed adhesion, or retraction. In the enthusiast space, Benchy, shown in Figure 2.14 is the quintessential benchmark part, although due to its age it is no challenging for desktop printers to produce a good Benchy. What makes Benchy a good benchmark part is the wide variety of features in a range of sizes: steep overhang, shallow inclines, holes, arches, thin features, cylinders. The features include 20 linear dimensions that can be measured to check dimensional accuracy using calipers, and 2 angular dimensions. The part can be used to identify a plethora of issues or defects mentioned before. Identifying these defects can help troubleshoot specific issues, like loose belts or excessive temperatures, thus a good benchmark part not only tests a printer's capabilities but also works as a troubleshooting tool.

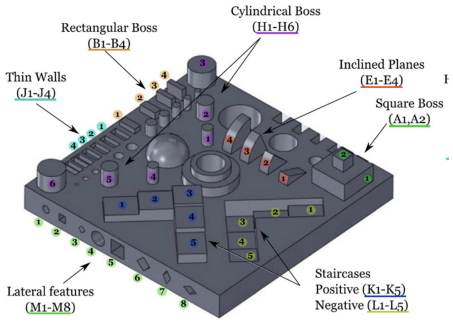
In academic literature there have been proposals for standardized FDM benchmark parts. In 2012 Moylan et al examined existing test artifacts and propose a part.[63] This work in elaborated upon by Sanchez et al.[64] A wide variety of test artifacts, some of which are shown in Figure 2.15, are reviewed by Rebaioli et al.[65] In the review a total of 60 publications proposing test artifacts are considered, between 1991 and 2017.



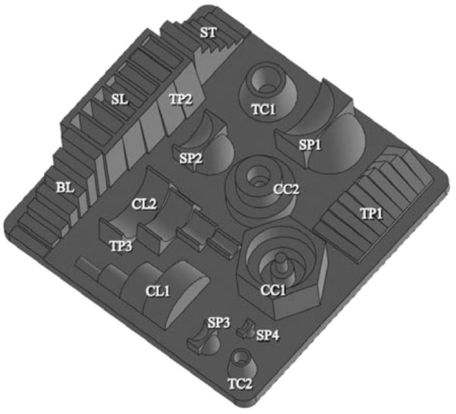
(a) Childs et al in 1994[66]



(b) Johnson et al in 1997[67]



(c) Sanchez et al in 2007[64]



(d) Minetola et al in 2016[68]

Figure 2.15: Various proposed benchmark artifacts reviewed by Rebaoili et al[65]

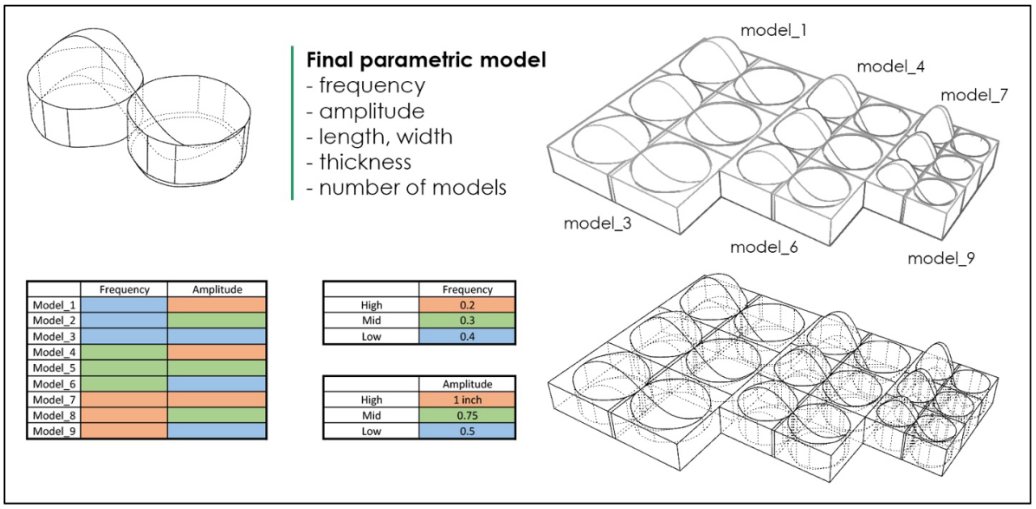


Figure 2.16: Benchmark part for conformal slicing by Jalui et al[69]

2.2.2 Non-planar benchmarking

Many of the previously stated benchmarking artifacts are designed specifically for conventional FDM, and as such focus on the capabilities of the conventional process. To understand what a benchmarking part for non-planar or multi-axis printing should include one must consider the unique capabilities of the method. Referring back to Section 1.1.3.1, the geometric benefits of multi-axis printing include the ability to print large overhangs without support structures, as well as reduced aliasing.

The first main feature that a multi-axis benchmark part should focus on is overhanging geometry. This should include various degrees of overhang, so that the printer can be evaluated on how well it deals with an increasing overhang. Even overhangs of more than 90 degrees - "underhangs" - can be present as they can be possible with a multi-axis setup. The second main feature is shallow angle surfaces. Conventional FDM processes struggle with surface roughness on these features due to the aliasing inherent to the layer height.

In 2022 Jalui et al developed a test artifact specifically for testing the anti-aliasing capabilities of conformal slicing, shown in Figure 2.16.[69]

2.3 PATH-PLANNING

2.3.1 Feature-based

Feature-based slicing treats different features of a part as individual structures, printing them sequentially, as shown in Figure 2.17.[70] E.g. if your part has many vertical features, they could be printed one after the other to reduce stringing between the features, as long as there is sufficient clearance between the features to ensure the hot-end doesn't collide. This approach requires more significant modifications to existing slicing software, as specific features of a part may have to be isolated to define the sequence of printing. Typical slicers don't have functionality to dis-

sect parts into their individual features. Alternatively, a general algorithm could be used to apply feature-based slicing to parts, as described by Jiang et al.[70]

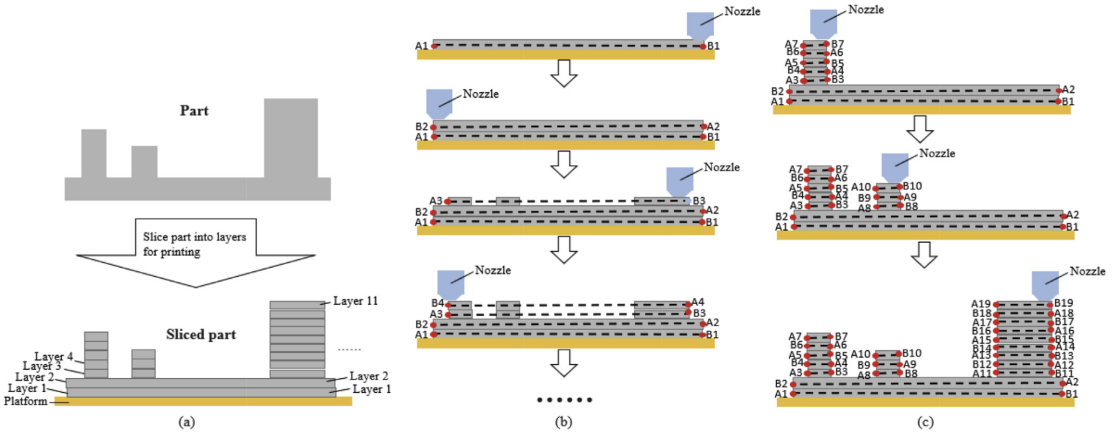


Figure 2.17: Feature-based slicing logic[70]

2.3.1.1 Multi-axis feature-based slicing

In 2014 Lee et al showed feature-based slicing implemented in a multi-axis workflow.[31] A portion of the part that does not have any overhanging geometry was sliced in the normal Z build-direction, then an overhanging feature is sliced separately in a different build direction, as shown in Figure 2.18. Using this approach, no support-structures are needed to print the overhanging geometry, as the build direction is normal to a surface on a precedent surface, acting as a build-plate for that particular feature. The part was printed using the 5-axis printer described in Section 2.1.4.2.

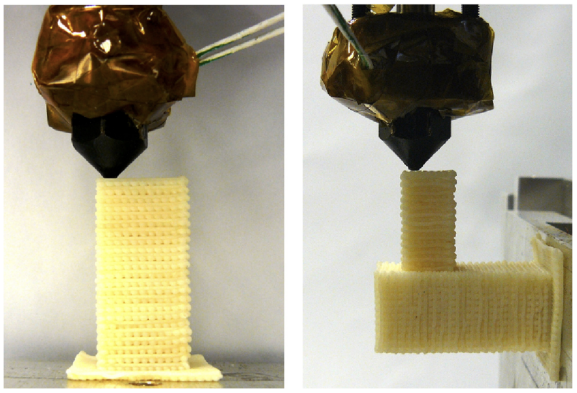


Figure 2.18: Multi-axis feature-based print with unsupported overhang by Lee et al[31]

Ding et al expands on this principle, with an algorithm for decomposing a part into features to be used for multi-directional slicing, as shown in Figure 2.19. The decomposed parts are sliced with a build-direction in which there is no overhanging geometry for that portion. All sub-volumes are later regrouped.[28] Murtezaoglu et al present a method for applying feature-based printing to parts without distinct features. Instead changing the build direction where the overhang angle would interfere with printing, as shown in Figure 2.20.[71] In 2020 Wu et al propose a generalized

algorithm for decomposing even complex geometry into the necessary portions to print without or with reduced supports, as shown in Figure 2.21.[72] This is a breakthrough in feature-based slicing, as even complex geometry can automatically be decomposed for multi-axis printing. Various example parts are shown and are successfully printed on a 5-axis robotic arm build-plate.

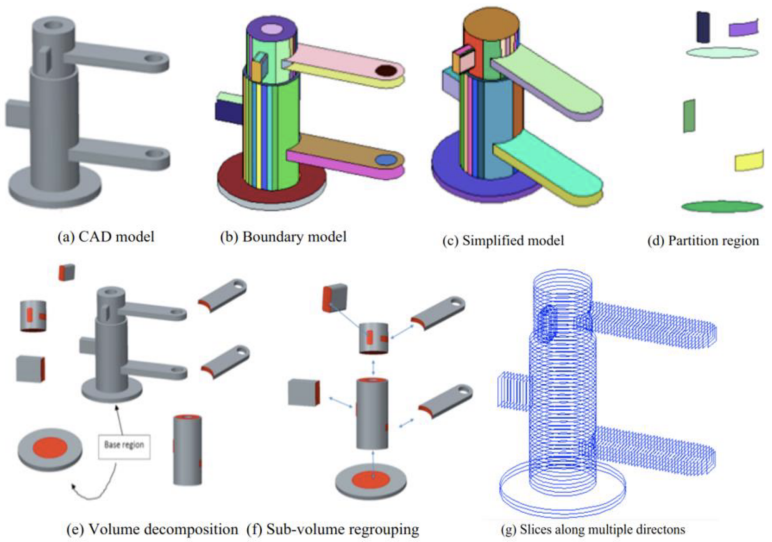


Figure 2.19: Multi-directional slicing principle by Ding et al[28]

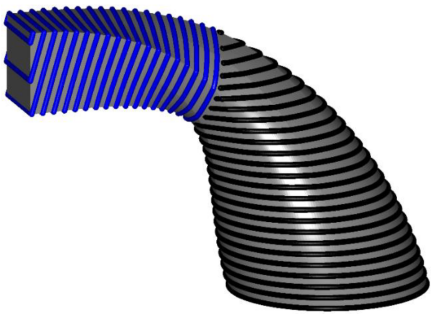


Figure 2.20: Multi-directional slicing by Murtezaoglu et al[71]

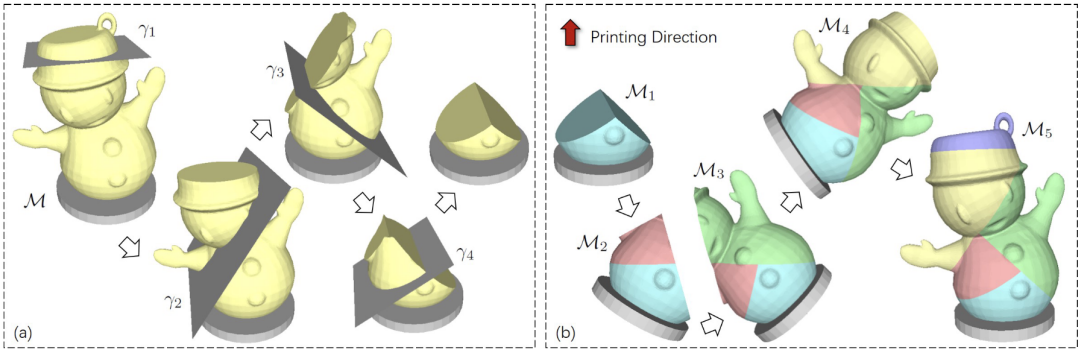
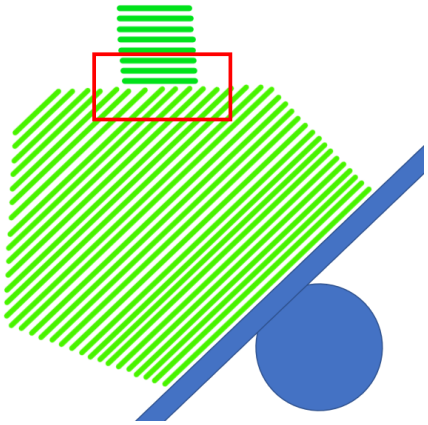
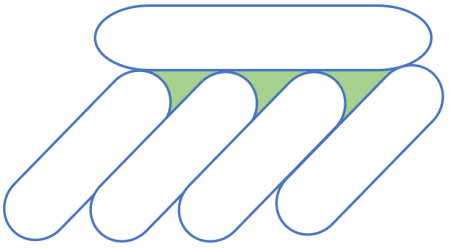


Figure 2.21: Generalized part decomposition for multi-directional printing by Wu et al[72]

A key challenge in multi-axis feature-based slicing is the uneven substrate for consecutive features. Due to the stair-stepping effect caused by the stratification of layers, side surfaces are rougher than those parallel to the build-plate, particularly at shallow angles. This presents a challenge in ensuring adhesion between the differently angled features, as shown in Figure 2.22.



(a) Second feature printed on oblique surface



(b) Voids, highlighted in green, causing porous interface

Figure 2.22: Challenge in adhering first layer of new feature to side surface substrate of existing part

2.3.2 Conformal and Curved

Conformal slicing, also referred to as Curved Layer FDM (CLFDM) is the concept of slicing the part not in flat parallel layers, but in curved layers, or layers that conform to a certain shape, as shown in Figure 2.23. The technique was first presented by Chakraborty et al. in 2008.[27] As can be seen in the schematic representation, it was initially conceived of as depositing curved layers that conform to a curved work table.[73] Naturally this clashes with a core principle of AM, as it would then require specific tooling for parts.

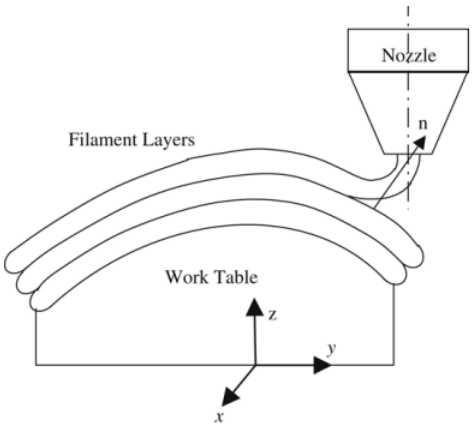


Figure 2.23: CLFDM concept by Chakraborty et al[27]

In 2015 Allen and Trask demonstrated the same principle applied with an internal lattice infill, as is common in the conventional FDM process. In their demonstration they also highlight the unique benefit of printing curved sandwich materials, which have a material change between the skins and the internal (honeycomb) lattice structure. Their demonstration is done on a typical 3-axis delta printer, illustrating that some of the basic non-planar printing can indeed be performed on typical printers, as long as clearances around the nozzle tip don't impede the toolpaths. In 2016 Lim et al showed a Grasshopper (GH) script for generation of conformal toolpaths.[74]

In 2018 Ahlers presented a method for conformal slicing that involved printing only the external surfaces curved.[17] In his method, a planar toolpath is projected onto the curved outer surface of a planar printed part, as showing in Figure 2.24. The method was implemented in a fork of the open-source slicer - Slic3r.[75] Thanks to the accessibility coming with the implementation in a mainstream slicer, Ahler's gained significant publicity as his work was published in an online video by Laws (better known under the alias CNCkitchen) in 2020, garnering over a million views.[76]

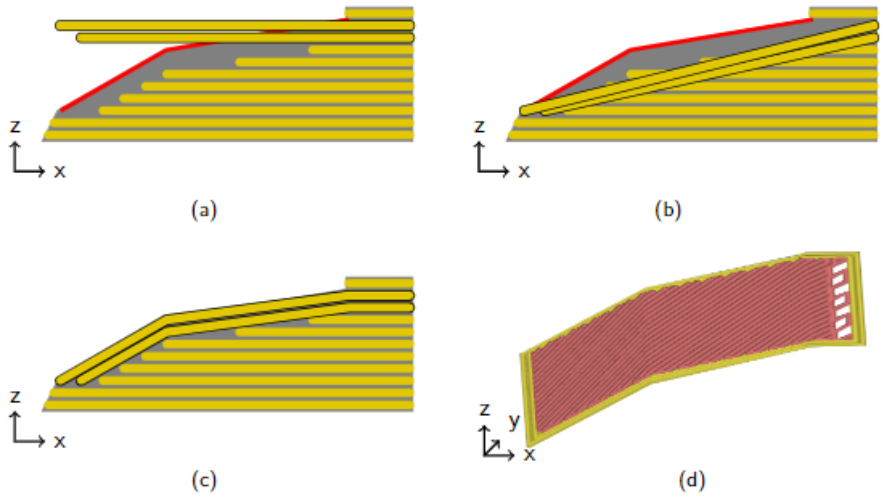


Figure 2.24: Ahlers method for mixed conformal slicing; (a) unprotected planar toolpath; (b) layer points are projected downwards; (c) projected layers are intersected with precedent layers; (d) toolpath of projected, curved layer[17]

Another approach for printing conformal layers without the need of a custom workpiece is the use of variable layer height. Zhao et al show this in their 2020 paper, referring to it as Variable Extruded Filament (VEF).[77] Conformal toolpaths have a high potential in AM of Continuous-Fiber Reinforced Polymer (CFRP) parts. As shown by Zhang et al, conformal toolpaths can be used to apply an outer perimeter of fiber-reinforced filament to a conventionally printed core volume, as shown in Figure 2.25.[77]

2.3.2.1 Isocurve layers

One approach for dividing a part into curved layers is using isoparametric curves, aka iso-curves. Iso-curves are curves on a parametric surface given by a function $f(u, v)$, for which one parameter (u or v) is fixed.[78] In practice, these can be thought of as the infinite amount of curves that make up a surface in one direction. Li et al have presented a way of using iso-curves to divide a

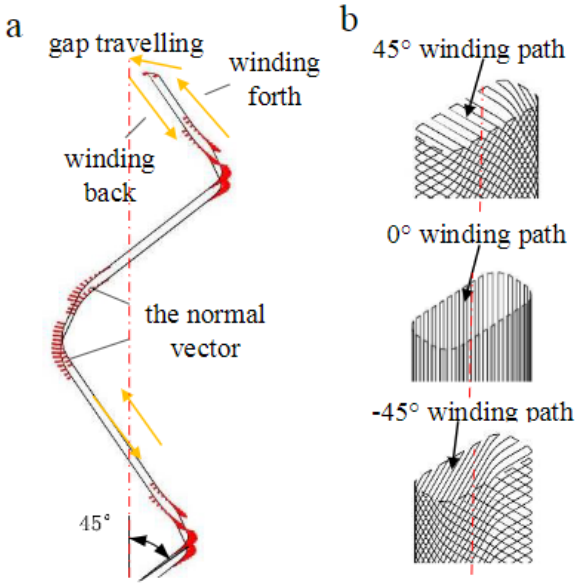


Fig. 3. (a) Diagram of 45° winding path; (b) three different geodesic winding trajectories around the core-volume.

Figure 2.25: Conformal toolpaths used for CFRP parts by Zhao et al; (a) single path conformed to core volume; (b) different conformal toolpaths transverse and at +-45°to build direction[77]

body into curved layers, as shown in Figure 2.26.[79] The β iso-curves can be used for defining layers, whereas the α iso-curves can be used for constructing internal structures. This iso-curve (or iso-layer) approach can be effectively used to slice an arbitrary part such that, if printed with the nozzle normal to the layer, overhanging angles are reduced, as shown in Figure 2.27. It is simple to implement, as iso-curves can be found for an arbitrary surface.

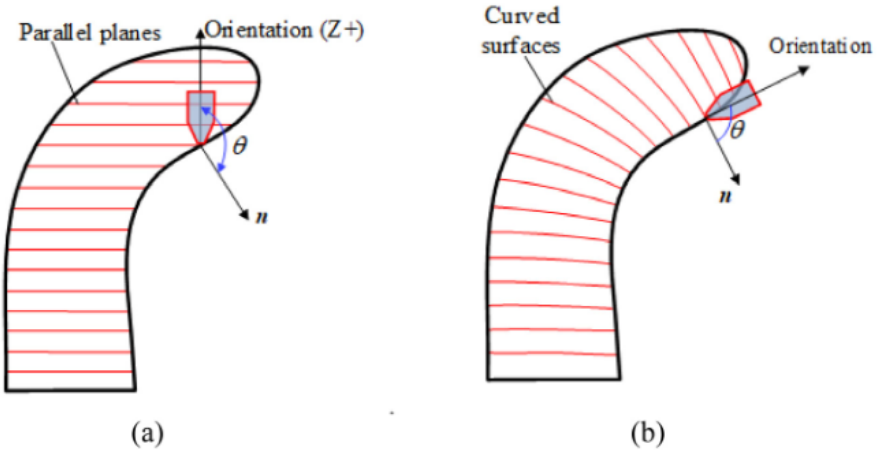


Figure 2.27: Effective reduction of overhanging angle using iso-curves with layer-normal nozzle[79]

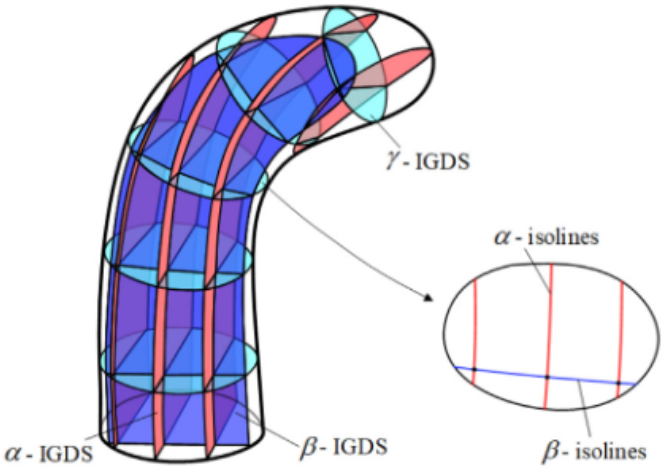


Figure 2.26: Isoparametric curves used to divide body into layers for printing[79]

2.3.3 G-code anamorphosis

Planar slicers, having been in constant development by the open-source community, are refined to a high degree, with excellent implementation of a generalized framework for slicing any part into perimeters, infills, support structures and more. Effectively making desktop 3D printers the most accessible CNC machines. So one argument goes, that an effective way to approach multi-axis path-planning is to add modification layers to existing planar workflows. In what could be called "G-code anamorphosis" the part model is distorted according to certain distortion parameters; that deformed part (projection) is then sliced into G-code using a typical slicing technique, after which the G-code is distorted using the inverse distortion parameters. This is a process analoagous to anamorphosis used in artistic expression, where the portrayed subject is drawn as a distorted projection, but then reprojected into it's true form, as shown in Figure 2.28.

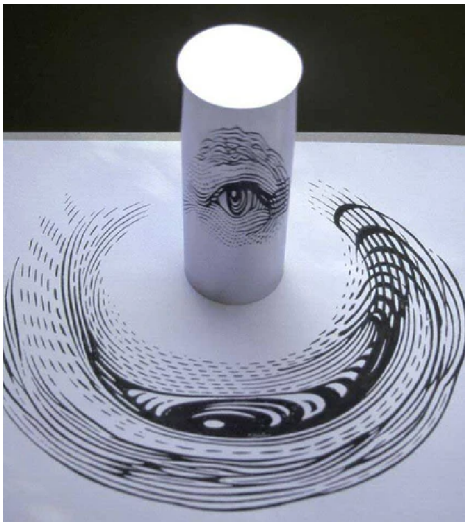
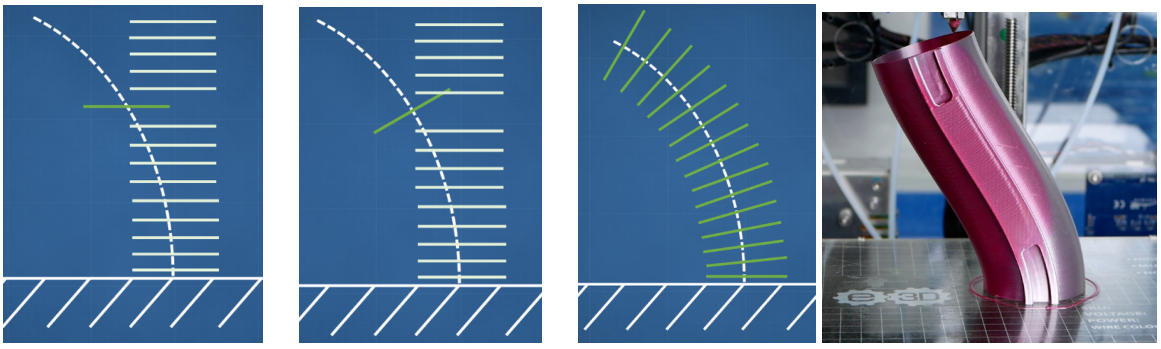


Figure 2.28: Example of mirror anamorphosis in artistic expression[80]

2.3.3.1 G-code bending

G-code bending, developed by Hermann in 2022, is the projection of G-code to fit a spline. Each layer of the G-code is moved translationally to the same height on the defined spline, then rotated to the tangent of that spline, as shown in Figure 2.29. The extrusion rate for each point is then recalculated to account for the variable layer height, using Equation 2.1, where $t_{nominal}$ is the layer height as sliced, and t_{local} is the defacto layer height after the rotational manipulation. This technique is useful for slicing parts with long curved features, like the one shown in Figure 2.29d, but is not suitable if the part has distinct features curving in opposite directions.[81]

$$flow_{local} = flow_{nominal} \cdot \frac{t_{local}}{t_{nominal}} \tag{2.1}$$



(a) Layer is translated to position fitting spline

(b) Layer is rotated to spline-normal angle

(c) The process is repeated for all layers

(d) Part printed using G-code bending

Figure 2.29: Working principles of G-code bending by Hermann[81]

Hermann executes this method using an open-source Python script.[82] The user defines the spline numerically using coordinates and boundary conditions. The script then takes a G-code file and performs the manipulation, outputting a modified version of that file, where point coordinates and extrusion rates have been altered to bend it.

2.3.3.2 Conical slicing

A different approach to G-code anamorphosis is conical slicing by Wüthrich et al.[56] Developed specifically for use with a 4-axis printer, described in Section 2.1.5, it can print overhangs up to 90°. To accomplish this, the part geometry is transformed conically before being sliced with a planar slicer. The resulting g-code is subsequently transformed conically, but inverted, as shown in Figure 2.30. While originally intended for use with the specific 4-axis slicer, the slicing approach can also be used with 3-axis printers as long as the angle of the cone is reduced, as shown by Hermann.[83] Naturally, it can also be used with 5-axis printers with one of the rotational axes fixed at the cone angle. Both a script for transformation of the mesh file as well as back-transformation of the G-code are open-source Python scripts.[84] The user can define the cone angle and cone center.

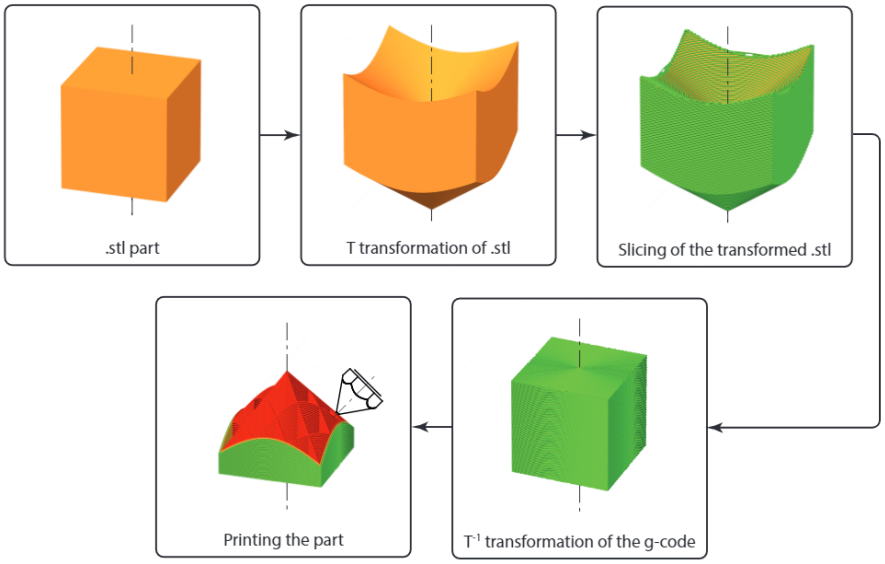


Figure 2.30: Working principles of conical slicing by Wüthrich et al. [56]

2.3.3.3 Free-form manipulation

The previous two G-code anamorphosis techniques were specific to certain types of modifying geometry - in G-code bending, the geometry is manipulated with a spline; in conical slicing the geometry is manipulated with a cone. The ultimate evolution of this principle is the "nozzleboss" G-code importer and re-exporter by Löpmeier. It is an add-on for Blender, which allows the user to import G-code as a mesh, and then perform any kind of manipulation with it that can be done to a mesh in Blender. The G-code can then be re-exported for printing. The simple user interface in Blender allows for intuitive modification of planar G-code into non-planar.[85]

2.3.4 General

The ultimate goal in path-planning is to achieve a generalized solution - that is, an approach that can be applied to any arbitrary part to achieve the desired results. So far we have examined specific path-planning strategies, each of which lends itself to certain kinds of geometry or use-case. Generalized path-planning strategies are those that could use any part geometry and slice it to a set of desired outcomes.

In 2018 Dai et al presented an algorithm that can universally slice parts support-free. They show a wide range of geometry, in Figure 2.31a, printed support-free using a 6 DoF robotic arm.[47] In 2020 Fang et al showed a generalized solution for printing strength-optimized parts, to counteract the anisotropy typical to FDM. The method could print parts with significantly improved strength in the user-defined load-case, shown in Figure 2.31b.[32] The two are examples of generalized single-objective slicers.

What could be considered the first general multi-objective multi-axis slicer is the S^3-Slicer presented by Zhang et al in 2022.[46] The presented algorithm adjusts individual tetahedron elements in a mesh to achieve one of three objectives: support-free (SF); strength reinforcement (SR); and

surface quality (SQ). The approach is truly general, as the user can define a combination of desired objectives, with specific weights for each, as shown in Figure 2.32.

(a) Generalized slicer for support elimination[47]

(b) Generalized slicer for part strength[32]

Figure 2.31: Generalized single-objective multi-axis slicers

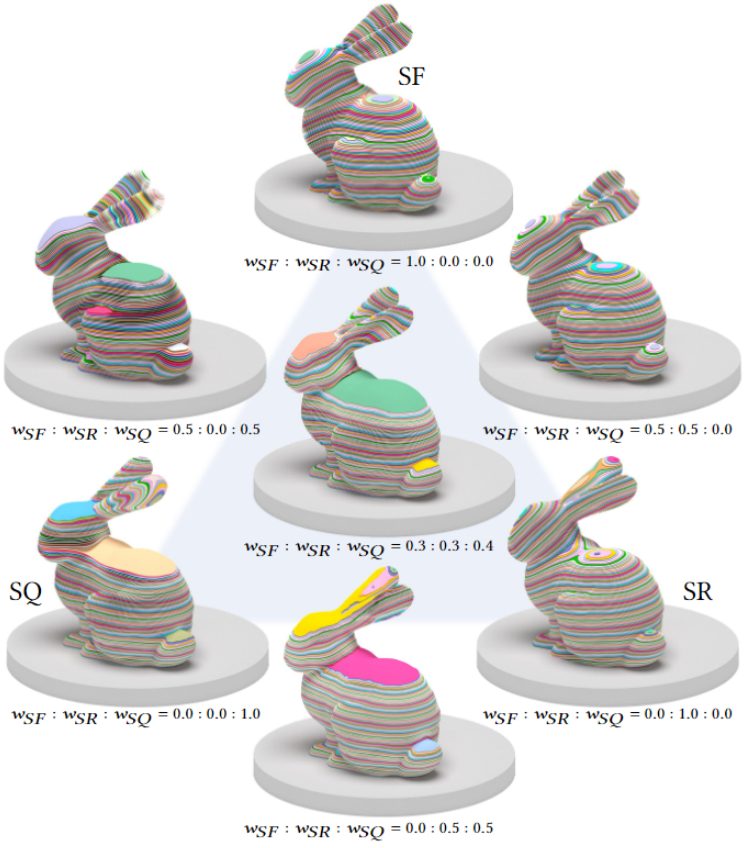


Figure 2.32: General multi-objective multi-axis "S3-slicer" by Zhang et al enabling control over extent of objectives[46]

2.4 DISCUSSION

In this chapter we delved into the state of the art in multi-axis additive manufacturing, to determine the most effective approach for achieving the research objective outlined in Section 1.3. First, we examined the solution space for hardware, with a focus on kinematic systems, which are the core of the task at hand. In order to meet the requirement of accessibility only combined 3+2 mechanisms are considered. The ability to retrofit existing 3-axis printers with a modification to enable 5-axis printing greatly reduces the barrier of entry to the technology, which is part of the goal. Out of the three configurations - table-table, table-head, and head-head - only those that isolate modifications to a single subsystem (head or table) should be considered for accessibility. The HH is favored due to its larger build volume and consistent accuracy,[5] which helps make the resulting printer design an appealing choice for results useful past just samples.

The history of benchmark artifacts used in AM, and FDM specifically, was reviewed. Artifacts designed specifically for multi-axis AM are limited. For the purposes of showing the full capabilities of multi-axis FDM, several different parts will be printed to test the dimensional accuracy, surface roughness, waste reduction, and overhang capabilities of the printer. Dimensional accuracy and surface roughness will be tested using 3D scanning. Due to the lack of a common benchmark, generic tubular parts with a wide range of overhang and surface angles will be designed for testing. Additionally, a variety of complex geometry parts will be used - both aesthetic and functional. Different sizes of parts will be used, to validate the printer's capability to print at the extremes of build-volume.

A wide variety of path-planning algorithms were identified in literature. Different kinds of geometry lends itself better to different approaches. While general multi-axis slicers have been presented, none are yet fully accessible. For the tubular parts iso-curves slicing will be used, as it should work well for the relatively simple geometry. G-code anamorphosis will be used for parts with more complex geometry for which iso-curves can not be generated, or that require infill.

Defining the general hardware requirements for the printer, the best way of benchmarking that hardware, and ways to prepare path-planning to execute those benchmarks, sets a good basis for fulfilling the research objective.

[5] This is further explained in Section 3.1.1, giving an analytical justification

2.5 RESEARCH QUESTIONS

Reflecting on the research objective, and what is shown in the state of the art, these research questions were formulated, given the gaps in existing literature:

- How does a multi-axis FDM process compare to a conventional one in terms of surface roughness?
- How much waste - material and time - can be reduced by using multi-axis FDM to print parts with overhanging geometry?
- How does a multi-axis FDM process compare to a conventional one in terms of dimensional accuracy?
- What design parameters of the multi-axis 3D printer are critical for achieving these benefits, and what can be done to improve results further?
- What path-planning algorithms can be used to print the benchmark artifacts?

Chapter 3

Mechanical design

IN THIS CHAPTER WE TURN OUR FOCUS from theoretical exploration to implementation. Firstly I reiterate the **design requirements** that were decided based on a combination of the motivation and examining the state of the art. These requirements dictate both the high-level architecture of the device, as well as more minor details of the mechanical design.

Based on the requirements, I delve into **concept development**. For both of the leading concepts I explain the full development process, which starts with *inspiration* for the concept, and *rationale* about what makes it a suitable candidate. The *implementation* of the concept is explained, examining the specific mechanical design decisions, and how they relate to the requirements. *Evaluation* of the concept prototypes lead to defining requirements for the following iteration. Finally, I explain how the print-head is integrated into the host printer.

3.1 DESIGN REQUIREMENTS

A key driver of the design requirements is accessibility. Thus, the designed hardware will be a system that can be retrofit onto existing FDM printers. This means that users can modify an existing printer with the 5-axis supporting hardware, to access the benefits with a low barrier of entry. This reasoning leads to the choice of using a 3+2 combined mechanism, which was thoroughly explained in Section 2.1.4. with that approach, the X, Y and Z axes of the base printer are augmented with two rotational axes. The additional axes were chosen to be applied to the head instead of the table, based on three main benefits observed from the state of the art - larger build area[1], smaller hardware size, and consistent error across the build-volume. In this chapter we delve deeper into this justification.

Another design choice that aids accessibility is self-replication. As was explained in Section 1.1.2.1, self-replication has been instrumental in the rapid growth of the FDM space, and including the principle in the development of this work ensures the community has easy access. This means that parts were designed to be printable using a common FDM printer,[2] excluding COTS parts, such as fasteners and bearings.

[1] Relative to printer size

[2] All parts were printed using Ender 5 Pro or Prusa i3 MK3 printers

3.1.1 Build area and volume

Adding additional axes to a FDM printer reduces its effective build area and volume[3], as the rotation of the build-plate or print-head will require additional clearance within the build volume. We can call this the clearance volume - it is the volume which the print-head requires access to in order to reach the extent of the build volume.

For a 5-axis printer, we can calculate the clearance volume using geometry. The calculation requires dimensional values for the build volume - width (w), length (l) and height (h) - as well as the location of the rotational axes relative to the build-plate or print-head. The A axis can be assumed to be centered on the build-plate or print-head. For the TT configuration the B axis can be assumed to be in the centroid of the build-volume, as has been applied in previous designs.[52][86] For head-head configurations the B axis can be assumed to be at the furthest end of the hot-end assembly.

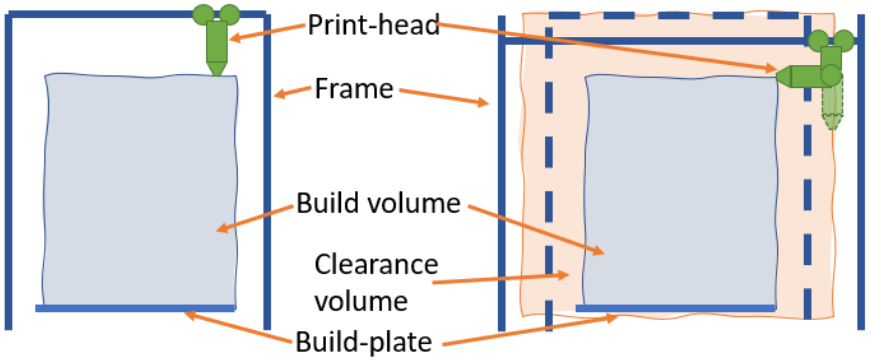


Figure 3.1: Increase in relative printer size to accomodate rotation of print-head

Figure 3.1 illustrates how the external dimensions of a head-head configuration printer must grow to maintain the same build volume. On the left is the front view of a printer, with a frame, a build-plate, and a print-head. The available build volume is indicated by the blue-shaded area. On the right the print-head has been actuated normal to the right face of the build volume. As can be seen, this requires the frame to be larger, relative to the old frame size shown (now shown in dashes lines). The orange-shaded area represents the clearance volume, that must be accessible to the print-head in order to reach the entire build volume.

$$V_{clearance} = (l + 2d_B) \cdot (w + 2d_B) \cdot (h + d_B) \tag{3.1}$$

$$V_{reduced} = (l - 2d_B) \cdot (w - 2d_B) \cdot (h - d_B) \tag{3.2}$$

The clearance volume of the head-head configuration can be modeled as follows. The clearance needed on each side of the build volume is the distance between the B-axis pivot point and the

[3] In traditional CNC terms, the build-volume would be referred to as machinable volume.

nozzle tip – d_B . To each dimension of the build volume we add this clearance times two, since the volume needs to be accessed from both sides, except for the height, as the bottom of the build volume cannot be accessed, as shown in Figure 3.1. The product of these three dimensions is the volume, as shown in Eq 3.1. Likewise, by instead subtracting d_B we can calculate the reduced build volume $V_{reduced}$, if you were to retrofit a printer with a 2 DoF head-head manipulator, as per Equation 3.2.

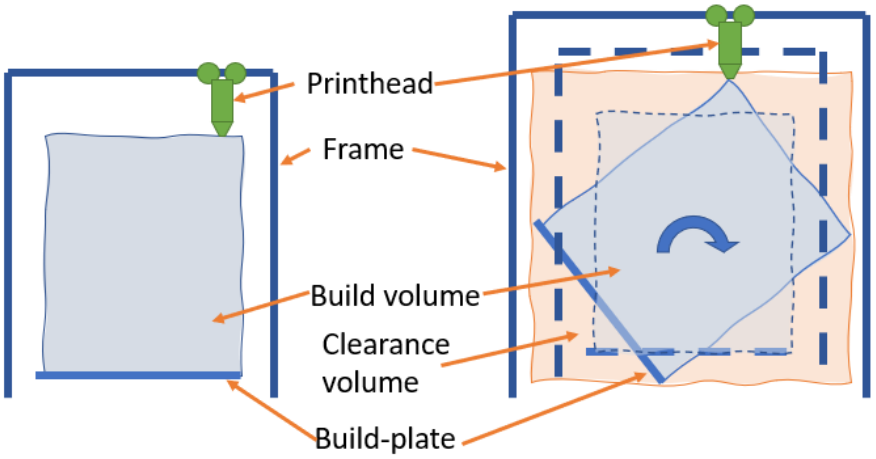


Figure 3.2: Increase in relative printer size to accomodate rotation of print-bed

Figure 3.2 illustrates how the external dimensions of a TT configuration printer must grow to maintain the same build volume. Like before, the left shows the typical printer. Then, on the right the print-bed has been rotated. You can tell from the blue-shaded area that it no longer fits within the old frame, which is now shown in dashed lines. Instead, the orange-shaded area represents the clearance volume now required.

$$V_{clearance} = \sqrt{\sqrt{w^2 + l^2}^2 + h^2}^3 = (w^2 + l^2 + h^2)^2 \tag{3.3}$$

The clearance volume of the TT configuration can be modeled as follows. The A axis enables the rotation of the build-plate around the centroid of its area. Thus the longitudinal dimension of the clearance area is the diagonal of the build area $d_A = \sqrt{w^2 + l^2}$. The B axis enables rotation of the build-plate around the centroid of the build volume, as shown in Figure 3.2. Thus the width, length and height of the clearance volume are the diagonal of the build height and the diagonal of the clearance area. Since all dimensions of the clearance volume are identical, they are cubed to get the volume, as shown in Equation 3.3.

$$V_{reduced} = \frac{min(l, w, h)^3}{3^{3/2}} \tag{3.4}$$

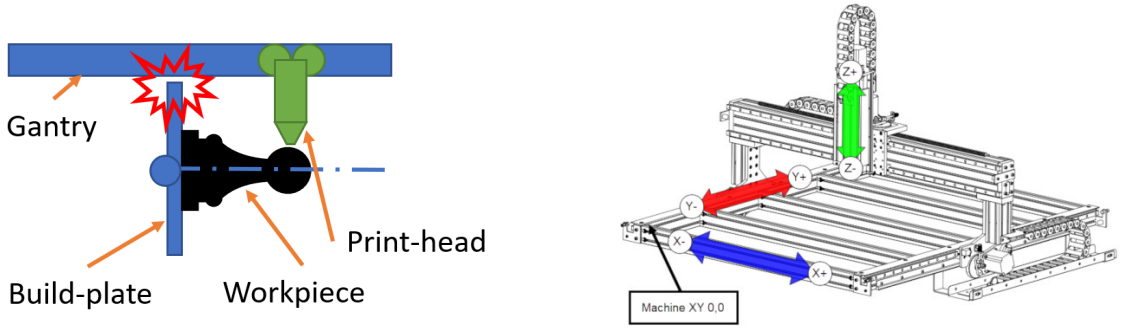
The real build volume of a TT configuration printer is a semisphere. Since the largest cuboid that can fit a sphere is an equilateral cube, thus the optimal build volume for a TT printer is an equilateral cube. Knowing this we can calculate the build-volume reduction from retrofitting an

Table 3.1 Comparison of reduction of effective build volume between HH and TT configurations of retrofit Ender 5 Pro and Plus

Configuration	*l*	*w*	*h*	*V_build*	*V_reduced*	*%*
			Ender 5 Pro			
Head-head	220	220	300	14520 cm^3	3467 cm^3	23.88%
Table-table					2049	14.11%
			Ender 5 Plus			
Head-head	350	350	400	49000 cm^3	8251 cm^3	43.81%
Table-table					21465	16.84%

existing printer with a 2 DoF build-plate. The diameter of the spherical build volume can be no larger than the shortest edge of the original build volume. Thus the reduced build volume can be calculated using Equation 3.4.

Using these calculations we can compare the difference in theoretical reduced build volume between a TT and head-head configuration, and numerically arrive at the same conclusion that was previously derived analytically. Table 3.1 shows the build volumes, and the respective reductions of them, for both configurations of the host printer Creality Ender 5 Pro, which will be retrofit. As can be seen, the head-head configuration appears to have a significantly larger effective build volume, a roughly 69% increase. Considering the hypothetical retrofit of the larger Ender 5 Plus printer, we can see that the larger size furthers this disparity. In general terms, the TT configuration doesn't scale well, as the clearance volumes scales exponentially, whereas for the TT configuration it scales linearly.[4]



(a) (b)

Figure 3.3: (a) Build area limit inherent to build-plate interference with X gantry; (b) portal configuration that maximizes clearance between X gantry and workpiece[87]

In addition to the aforementioned loss of build volume, TT configurations also face another challenge associated with larger build areas - interference between the build-plate and the X-axis gantry. As shown in Figure 3.3a, there is a collision hazard at high B-axis values, depending on the exact geometry of the hot-end. In principle it is not a barrier, as the issue is simply solved by using a so-called "portal" configuration, common for CNC mills, where the Z-axis is the final link of the kinematic chain, as shown in Figure 3.3. However, such configurations are extremely uncommon in FDM machines, as there is typically no advantage of it.

[4]There is a theoretical break-even point, below which changes to the build volume do not impact the frame size.

3.1.2 Hardware size

The previous section covered how the TT configuration makes poor use of the volume of a printer. But the size discrepancy between build-plates and print-heads alone should be considered a significant penalty. Considering the intent of self-replication, requiring to print large parts is a disadvantage, and reduces accessibility. The extra time expenditure raises the barrier for a curious maker to try it out. E.g., the largest part on the Open5x design has been designed to be just small enough to fit the Prusa i3 build-plate.[86] But that doesn't take into consideration that printing such large parts has additional challenges with regards to warping. The significantly higher print time, alongside the increased risk of print failure makes printing such large parts less attractive. While the Open5x design has been iterated upon, and a smaller variant for a different printer exists, the TT configuration will nonetheless have the disadvantage in part sizes for build areas that are comparable to what is expected from desktop FDM printers. On the other hand, the head-head configuration can have smaller parts, as the hot-end itself is smaller than the build-plate. Additionally, the size of the assembly doesn't scale up with the size of the build volume, as it does for actuated build-plates.

3.1.3 Kinematic error

The addition of the rotary axes reduces the accuracy of the printer. As more links are being added to the kinematic chain in series, the errors of those individual links will contribute to the overall kinematic error. For rotational axes, the rotational accuracy translates to linear accuracy depending on the lever distance of the measured point to the axis. Thus the farther the point in question is from the axis, the higher the error, unlike linear axes, for which the error remains consistent throughout its entire actuation range. The relationship can be stated using the Pythagorean theorem, as shown in Figure 3.4. The rotational error α translates to the tolerance of b,[5] or an accuracy of $\pm\frac{b}{2}$, which can be calculated as per Equation 3.5.

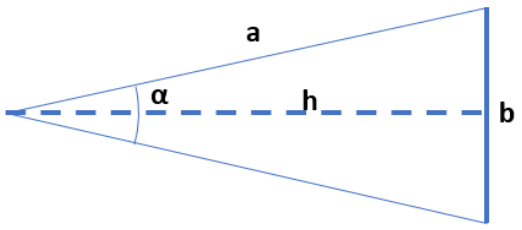


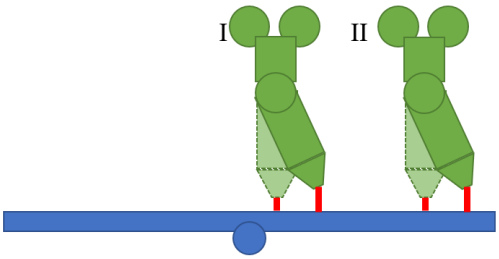
Figure 3.4: Diagram of calculating error of rotational axis at given lever arm

$$\frac{b}{2} = tan(\frac{\alpha}{2}) \cdot h \tag{3.5}$$

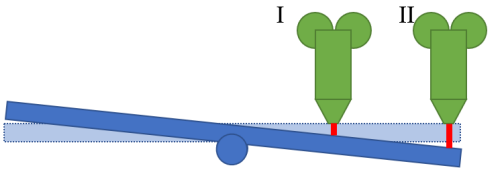
Figure 3.5b shows how an error in the rotation of the build-plate leads to a higher error towards the extreme of the build-plate. Thus, the larger a part is printed, the greater the error. On the other hand, Figure 3.5a shows how an error in the rotation of the print-head is dependent on the

[5]Simplification, as the b would actually be an arc, but at the low α values the difference is negligible

distance between the axis pivot point and the tip of the nozzle. Thus the kinematic error of a TT configuration is dependent on workpiece size, and scales linearly with it, whereas for the head-head configuration that error is constant.



(a) Illustration of how the error on HH configuration is decoupled from lateral print-head position



(b) Illustration of how the error on TT configuration is greater towards the build-plates extremes

Figure 3.5: Comparison on impact of printhead position on kinematic error caused from rotational axes

To give an idea of the severity of this effect, using Equation 3.5 we can calculate that given the angular accuracy of typical stepper motors used in FDM printers, assuming typical microstepping of 16 micro-steps[6] per step, and assuming a typical build-plate lateral dimension of 220 mm, the linear tolerance would be a significant 0.054 mm.

When comparing the accuracies of six different 5-axis CNC machine configurations, My et al. found that the error of TT machines (referred to as Types I and II in the work) is significantly larger than that of HH machines (referred to as Types III and IV).[88]

3.1.4 Specification

Based on the requirements of larger build area, smaller hardware size, and higher accuracy, an analysis was made regarding how exactly those requirements are impacted by design choices. Firstly, the intent to develop the solution as a retrofit for existing printers was stated. Further, the choice between a TT and HH configuration for the printer's kinematic architecture is a crucial, as it greatly impacts all of the requirements, and even more, as described by My et al.[88] On all considerations, the HH prevails as the superior configuration.

3.1.5 Materials

Due to direct contact with, and close proximity to, the hot-end, the rail was printed with a higher-temperature polymer. The use of polycarbonate (PC) blend material offers a higher heat resistance up to 113°C[7] while still possible to print on consumer-grade desktop FDM printers, such as the Prusa i3 MK3.[89] Other parts were printed in Polylactic acid (PLA), which is a the most common

[6] Microstepping is a technique of stepper motor control that allows increased angular resolution at the expense of torque, commonly used in FDM printers.

[7] Heat Deflection Temperature under low load (0.45 MPa)

FDM material thanks to its low melting point and relatively low thermal expansion coefficient. In later prototypes all parts were printed in PLA, the hot-end didn't seem to transfer a substantial amount of heat. Dropping PC from the Bill of Materials (BOM) improves accessibility, as it is difficult to print on the cheapest printers.

3.2 VIRTUAL PIVOT

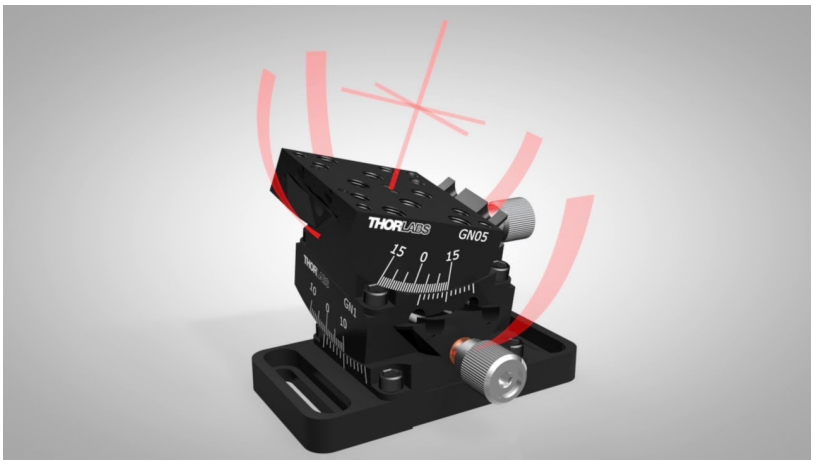


Figure 3.6: Goniometric stage by Thorlabs[90] as inspiration for print-head design

3.2.1 Inspiration and rationale

Inspired by the goniometric motion stage, shown in Figure 3.6, a virtual pivot mechanism was devised as a candidate for the print-head design. A goniometric stage is a device used to precisely rotate an object, with a point of rotation that is outside of the boundaries of the device itself. This is useful for rotating objects without displacing them. It can also be called an instant center device, as the instant center of rotation is a point that has no velocity on an otherwise moving object. Applying such a mechanism to the B-axis of the print-head would in theory allow the mechanical rotation of the nozzle's tip around this virtual point in space. Two main advantages were envisioned with this approach - elimination of IK, and retention of full build-volume.

3.2.2 Implementation

To achieve the instant center of rotation, goniometric stages utilize an arc-shaped track, that the carriage can travel on. In a goniometric stage this would be a high-precision motion system, and the driving the motion would be a screw. For the purposes of this project, this had to be simplified. The solution was the use of a custom V-slot motion system. V-slot in an open source project OpenBuilds, and was originally crowdsourced on Kickstarter in 2013.[91] The concept builds on the idea of T-slot rails, which are extruded profiles with a T-shaped groove that can be fastened to at any point along the length of the profile without modifying the rail. The versatility of these rails makes them a common choice for a construction element in industrial automation and research devices. OpenBuilds adapts this concept by adding a V shape to the groove, which also enables

special wheels to self-center in the groove. Carriages with several wheels can then be used directly on the extrusion, effectively turning it into a motion system. The big benefit of using V-slot rails is that they serve two functions at the same time - structure and motion system.[92]

The use of V-slot profiles is prevalent in desktop FDM printers, best exemplified by their use in the highly popular Ender series of printers by Creality. On Ender printers the frame is constructed using V-slot rails, and they are also used as linear rails for the print-head movement. Thus it was explored how a V-slot rail could be used for the B-axis. To suffiently constrain the movement of a carriage on an arc-shaped V-slot rail three wheels would be needed, as shown in Figure 3.7a. To also facilitate the drive of the axis, an arced gear-rack was used. So the inner wall of the V-slot rail was changed to an internal gear, and the corresponding wheel was changed to a spur gear, as shown in Figure 3.7b.

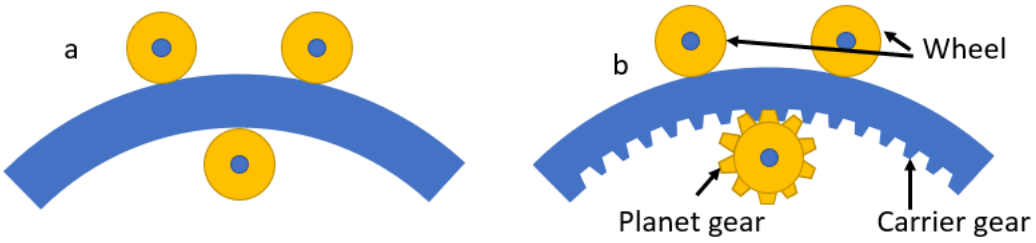
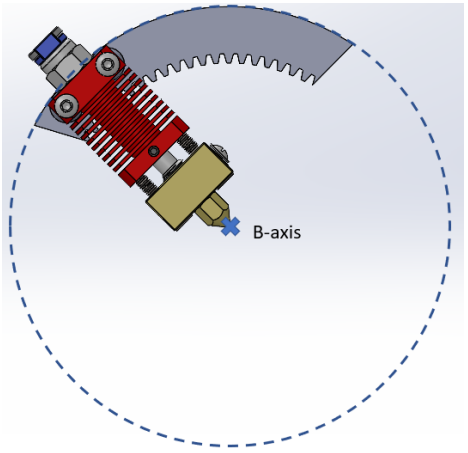


Figure 3.7: Use of arced rail in B-axis construction: (a) arced rail constrained by 3 wheels; (b) arced rail with gear rack, constrained by 2 wheels and a gear

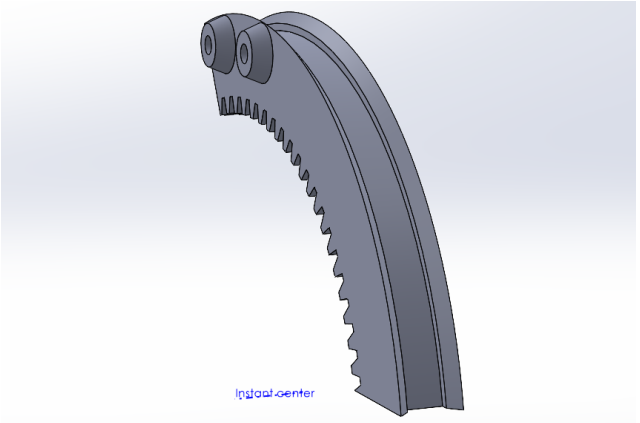
The size of the arc was based on the size of the Ender hot-end. The mounting points for the hot-end should fall somewhere on the rail, so an internal gear size was picked that was slightly smaller than the distance between the hot-end's mounting points and the nozzle tip (on the hosting Ender 5 Pro printer)[8][93], as shown in Figure 3.8a. The resulting internal gear is an 80 tooth gear, and a module of 1 was used for the sake of simplicity. The smaller spur gear was arbitrarily chosen to have 12 teeth.[9] To use terms analogous to what is used in epicyclic/planetary gearing, we will call the internal gear the carrier gear, and the smaller spur gear the planet gear (as denoted in Figure 3.7). The planet gear is driven by a stepper motor, and it is directly mounted on its shaft. The wheels (which have integrated bearings) and the motor are attached to a simple part (called the A-axis lever), which offsets the nozzle's tip to place it at the center of the A-axis rotation, as shown in Figure 3.9a. That lever arm is mounted directly on the shaft of the A-axis motor, which in turn is mounted to a part that fits directly to the existing X-gantry carriage of the Ender. The entire assembly that enables the additional 2 axes, shown in Figure 3.9b, contains just 3 printable parts, and COTS wheels, gear, motor and fasteners.

This assembly executes the fundamental kinematic purpose of the A- and B-axes, however, it excludes a number of important features. To position the motors, they should undergo a homing sequence, during which their accurate location is ascertained. This is normally done using end-switches or limit-switches, at one of the limits of the axes motion. The axis is then actuated until it hits that limit-switch, which serves as a home or reference position. For the first prototype no end-switches were designed, as it was deemed sufficient to manually set the home position. The

[8]The Ender 5 Pro hot-end design used in the design of the print-head is open source
[9]For gear choice the Mädler catalog (maedler.de) was used.



(a) Illustration of rail size picked to suit hot-end size



(b) Design of combined arc rail and (internal) gear rack - carrier gear

Figure 3.8: Carrier gear

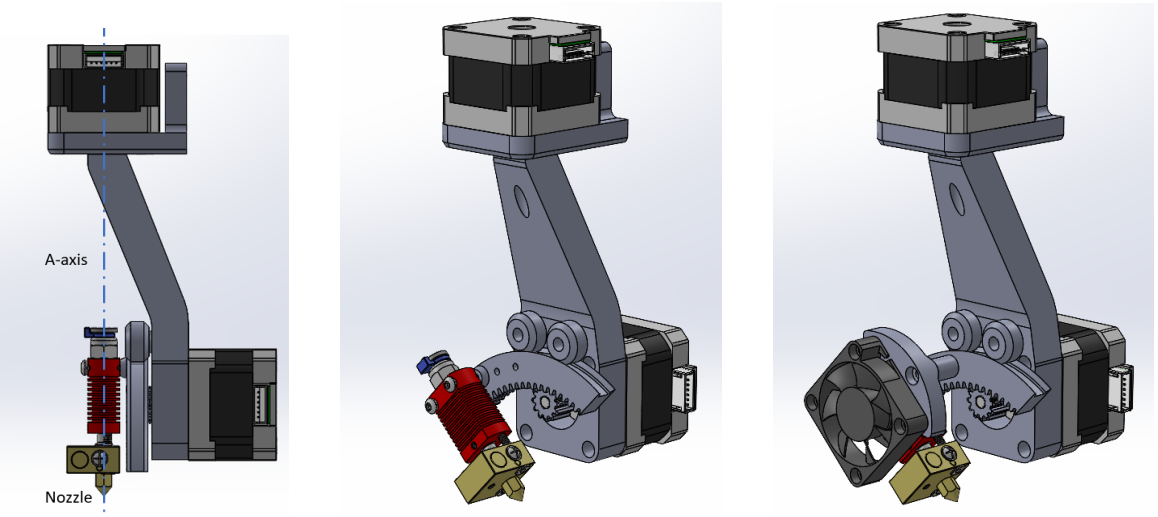
Table 3.2 Theoretical parameters of A- and B-axes of VP1 prototype

Axis	*Step size*	*Microsteps*	*Reduction*	*Accuracy*
A	1.8°	16	1	0.113°
B			80:12	0.017°

hot-end also typically houses two fans - one for cooling the hot-end's heatsink, and one for cooling the part. Neither were designed on the first prototype. The lack of a hot-end fan was, however, hindering initial tests, as it leads to clogging in the hot-end from heat creeping up the heat-break and melting too much filament, as explained in Section 1.1.2.2 (Working principles). To mitigate this effect, a part was designed to mount the Ender's hot-end fan to the carrier gear. The part cooling fan was not necessary for initial tests, as the convection from the surrounding air is sufficient to cool down the molten polymer exiting the nozzle, as long as print speeds are not fast, though some printing quality could be improved with forced part cooling. The assembly that was prototyped, including the hot-end fan, is shown in Figure 3.9c. It is named VP1 - Virtual Pivot 1.

The chosen carrier and planet gears produce a mechanical reduction of 80:12, or 6.67. Using a generic stepper motor of the Nema 17 size, which is the same size as used for all other axes on the host printer. Those motors have a step size of 1.8°, but generally use a 16 microsteps by default. Using Equation 3.6 we can calculate the angular accuracy of the axes. The holding torque of the axes is calculated as the product of the holding torque of the motors and the gear ratio of that axis.

$$Accuracy = \frac{StepSize}{MicroSteps \cdot Reduction} \tag{3.6}$$



(a) Offset of mechanism to align nozzle axis with A-axis

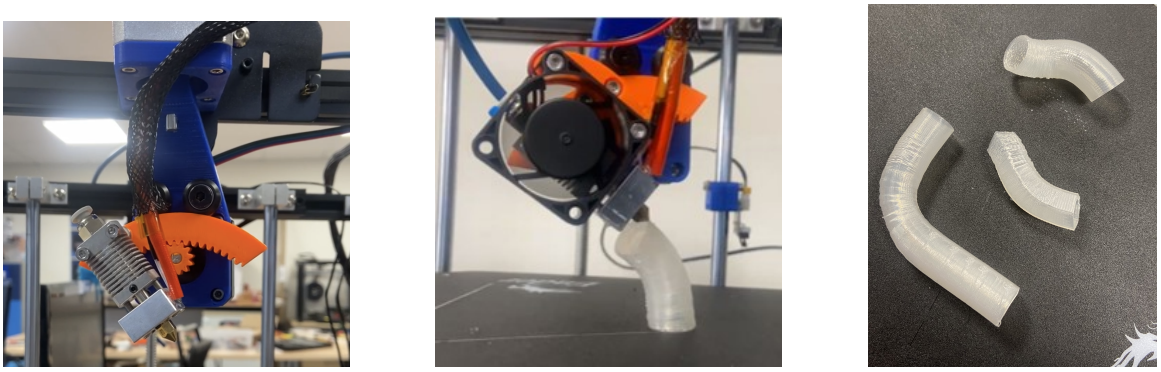
(b) Barebones virtual pivot assembly

(c) Virtual pivot assembly with hot-end fan - VP1

Figure 3.9: Virtual pivot concept for print-head

3.2.3 Evaluation

The VP1 prototype was critical to the initial set-up of the printer, as much core functionality of the Electronics and Firmware could not be validated until the first hardware prototype was ready. Such initial tests included "dry runs" using the barebones version, shown in Figure 3.10a, in which print jobs were executed without extruding any material, to ensure that the kinematics of the machine behave as expected. Unexpected behavior with a hot nozzle can (and has) caused damage to the print-bed. After installing a hot-end fan as shown in Figure 3.10b, further tests included 5-axis prints using Iso-curves slicing, explained in Section 4.2. Initial tests of such prints were successful without any particular calibration of the hardware, and could produce swept profiles with large unsupported overhangs, as shown in Figure 3.10c.



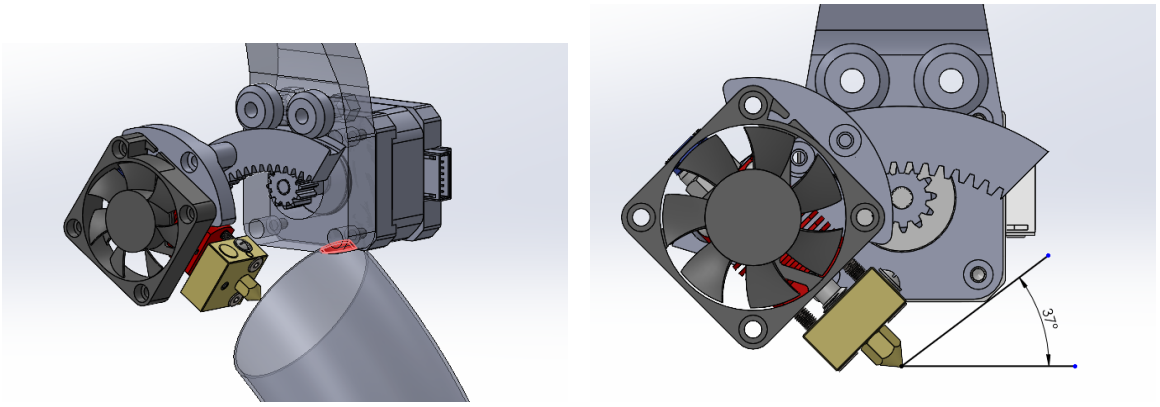
(a) First prototype VP1, barebones version

(b) VP1 prototype printing overhanging geometry

(c) Swept profiles printed with VP1

Figure 3.10: Prototype VP1

Attempts to print swept parts with a larger profile, as well as early attempts to print for strength reinforcement, highlighted the limited clearances around this print-head design. The interference between the workpiece and the A-axis lever, shown in Figure 3.11a, was an expected issue due to the low mounting of the B-axis motor. With the initial design, there is full clearance for B-axis angles up to about 37°, as shown in Figure 3.11b. With a permissible extrusion overhang of 45°, this printhead could print unsupported overhangs up to 82°unobstructed by interference. However, this clearance can be improved with a different design, discussed next.



(a) Interference between hypothetical workpiece and A-axis lever on VP1, shown in red

(b) Theoretical range of B-axis motion with full clearance

Figure 3.11: Clearances in VP1

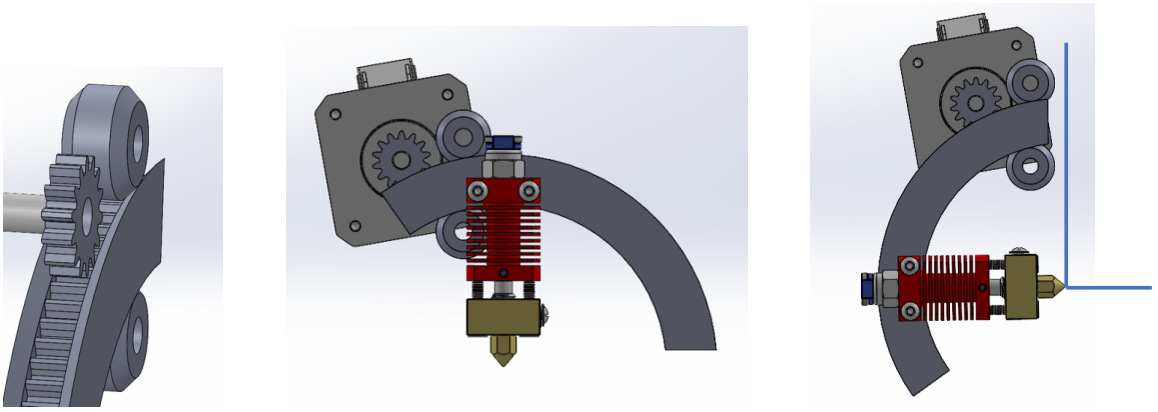
3.2.4 Improved design

Due to limitation of clearance between the workpiece and print-head, the design would only have full clearance at B-axis angles up to 37°, limiting overhangs to 82°. Returning to Figure 3.11 it can be seen that this interference happens with the B-axis motor, or the part of the A-axis lever that the B-axis motor is mounted to. Thus, there are two approaches to solving the issue: reducing the size of the motor; or moving it. Reducing its size would require a deviation from the chosen motor, which is very typical to desktop FDM printers.[10] As can be seen in Figure 3.11b, the motor could be moved either horizontally or vertically to improve clearance. However, the vertical position of the motor is constrained by the planet gear needing to be tangent to the inside perimeter of the carrier gear, as shown in Figure 3.7b. The solution is to move the gear to the position of one of the idler wheels on the carrier gear's outer perimeter. However, this required the outside perimeter to house *both* a gear rack and V-slot rail.

Integrating the gear rack and V-slot rail is shown in Figure 3.12a, where the gear is surrounded on either side by the V-shaped rails. Moving the driving planet gear to the outside perimeter naturally also permits its movement to the left of the two the available positions. By further shifting the wheels a bit, as shown in Figure 3.12b, full clearance space between the workpiece and motor section of the A-axis lever can be achieved at a B-axis angle of 90°, as shown in Figure 3.12c, pushing the possible overhang angle to 135°. It should be noted that there is still a possible interferance between the workpiece and the A-axis lever, as shown in Figure 3.13a, but it is at more than 100

[10]Using a typical Nema 17 size stepper motor is great for accessibility, as 3D printing hobbyists will definitely have spares of those motors

mm distance. Alternatively, a full clearance could be considered at a B-axis angle of about 81.5°, as shown in Figure 3.13b, but at that point there could still be interference with the X-axis gantry. Ultimately, there is no way to guarantee clearance, unless a kinematic system with a Z-axis column or "portal" is used, as described in Section 3.1.1 (Build area and volume) and shown in Figure 3.3b. The full design is shown in Figure 3.13c. On initial tests of the prototype, shown in Figure A.1, it appeared to have significant compliance at high B-axis angles due to the long arc of the carrier gear, and possibly the new arrangement of planet gear and idler wheels supporting the carrier gear. The concept was not further explored, and an alternative approach altogether was sought, explained in the following section.

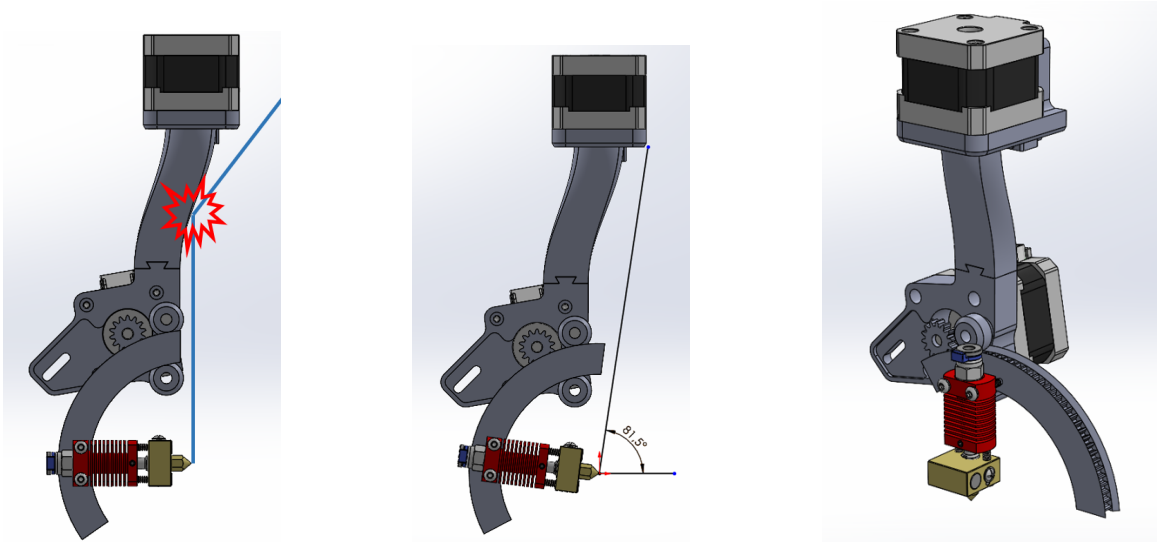


(a) Carriage gear prototype, with V-shaped rails

(b) VP2 layout with hot-end, carriage gear, planet gear and idler wheels

(c) Clearance of VP2 layout

Figure 3.12: Improvements made to VP2 to increase clearance with workpiece



(a) Interference location between workpiece and A-axis lever at B=90°

(b) Clearance up to X-axis gantry at B=81.5°

(c) Full VP2 design

Figure 3.13: Prototype VP2

3.3 ROBOT WRIST

3.3.1 Inspiration

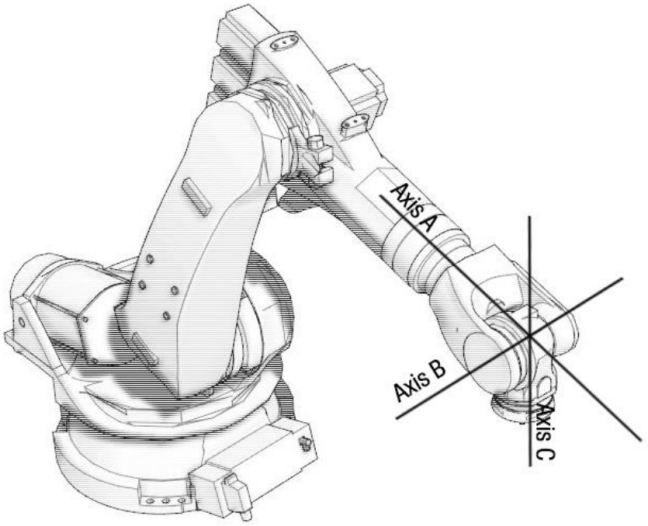


Figure 3.14: Robot arm, with wrist and hand axes shown[94]

The wrist of a robotic arm is considered the final collection of joints on the arm. The A and B axes, as shown in Figure 3.14, perform a function analagous to that of the human wrist - it rotates the EE/hand around it's central axis, and pivots it. These are the movements that we want to add to our 3-axis host printer. As was discussed in the State of the Art of printer kinematic systems, there are few implementations of the HH configuration for FDM printers, only one of which is applied as a retrofit to an existing printer.[55] The shortcomings of that design has guided decision-making in this design. I delve into the details later in this chapter, in Section 3.3.3.

3.3.2 Rationale

The benefits of the more traditional layout of axes, as opposed to the virtual pivot, described previously, are related to the clearance and stiffness of the mechanism. Even though a high clearance design was achieved with the VP2, it required sacrifices in the stiffness of the design, as the long and intricate carrier gear was qualitatively deemed too flexible to guarantee sufficient accuracy.

When comparing to the first concept, using the virtual pivot, two disadvantages can be considered - the reduction in effective build-volume, and the requirement for Inverse Kinematics (IK). Firstly, the reduction in build volume is unfortunate, but considering that as a HH configuration it already has a leg up against its TT counterparts in the build-volume department, so it's a reasonable sacrifice. Regarding IK, since a closed-form solution can be found, the IK solution is trivial, and would simply have to be implemented somewhere in the software stack. This is explained in detail in Section 4.5. Finally, this approach benefits from a higher degree of flexibility to adapt to any hot-end. The virtual pivot mechanism must be tailored to the specific hot-end of the host printer, as the different dimensions will impact the mounting location of the hot-end. Even just changing

the nozzle size on the virtual pivot may require an adjustment of the design. On the other hand, the robotic wrist can be adapted to generally any hot-end by simply defining the offset between the nozzle and the axes in the IK formulation.

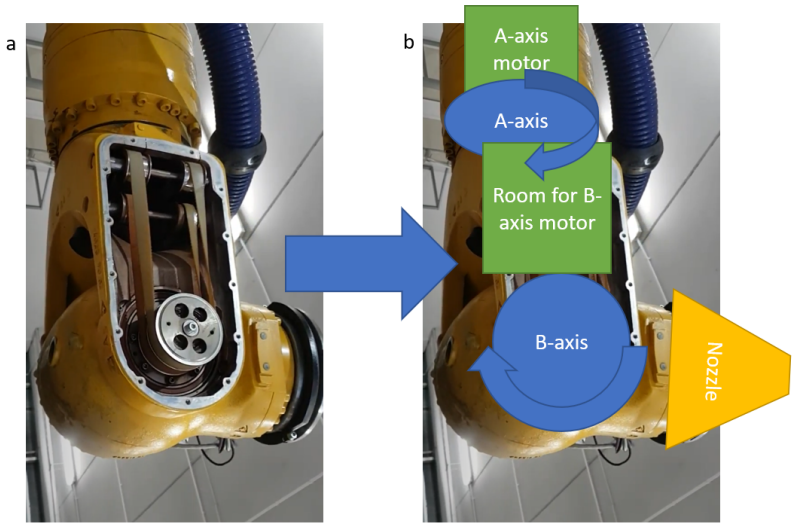


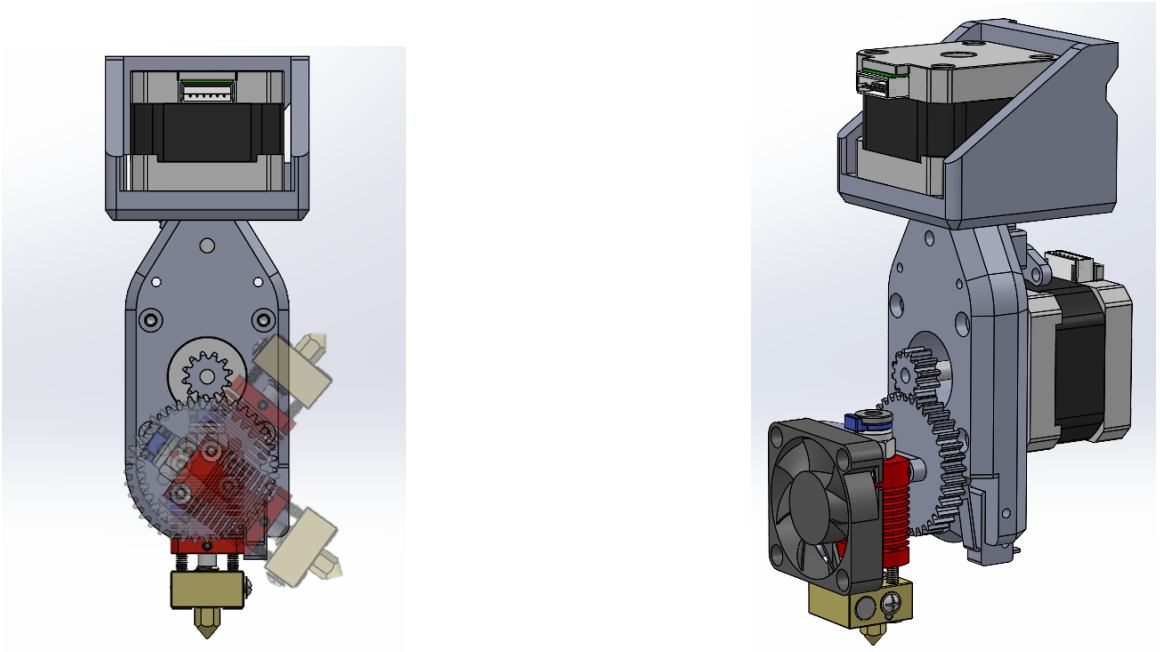
Figure 3.15: Robotic wrist used as basis for print-head design: (a) Kuka KR 500 wrist; (b) Proposed print-head layout overlaid

3.3.3 Implementation

When implementing the robotic wrist concept into the initial design a number of design decisions were made. The B-axis drive was decided not to be direct. This would allow the motor to be placed further away from the B-axis, to improve clearance around the nozzle. This also meant that, as part of the drive, a reduction ratio could be implemented. From experience with the VP1 prototype, the Bowden tube puts significant strain on the B-axis at extreme angles, thus the mechanical advantage of a reduction ratio would be beneficial to ensuring no steps are skipped by the motor. Looking at the construction of the forearm and wrist of a Kuka KR 500, shown in Figure 3.15a, we can see that the forearm has been designed to hold the B-axis motors, which then actuates the B-axis through a belt drive. The print-head design involves this same layout, as shown in Figure 3.15b, which reduces bulkiness around the B-axis.

Two drive solutions were considered for the B-axis - gear and belt. Each with their distinct benefits and drawbacks. The first benefit of the gear drive is that it could easily be prototyped. The gears can simply be printed on an FDM printer. While a belt drive can also be printed, it is a more involved process, as the belt would need to be printed out of a flexible material, such as TPU.

A disadvantage of the gear drive is the increased backlash compared to a belt-drive. Particularly for straight-cut gears, there is a possible dead-zone in the gear engagement, where play can be present. This backlash is detrimental to the accuracy of the axis. It can partly be remedied by using helical, herringbone or double gears. Those are typically undesirable solutions, as the added complexity adds to cost, even though printing of those gears is significantly easier than traditional cutting techniques. For the first iteration, straight spur gears were used without any gear sectoring. The B-axis has ample clearance, as shown in Figure 3.16a, to the extent that the full clearance angle



(a) High clearance in a wide range of B-axis motion

(b) CAD model

Figure 3.16: Prototype RW1

previously measured for the VP1 prototype is an irrelevant metric, as interference between the workpiece and hot-end would only occur at such extreme angles that interference with the X-axis gantry would be a greater issue. The CAD design is shown in Figure 3.16b.

3.3.4 Evaluation

The RW1 prototype proved capable of executing prints even without particular calibration steps. Moreover, the improved clearance allowed for printing of significantly larger parts, that were previously impossible due to collision risk between the print-head and workpiece. The first larger and more complex parts, such as the manifold part shown in Figure 3.17, were printed on this iteration. Through the testing of this iteration, the prior discussed issue of backlash became apparent. Under certain circumstances there was no issue, as the strain of the Bowden tube would keep the driven gear loaded in a certain direction for the duration of the print. This in fact acted like a spring-loaded anti-backlash system. However, at a certain B-axis position of about -45°there was an inflection point where the tube's strain would flip towards the other direction. Repeatedly, there would be a layer shift at about the -45°B-axis position, as shown in Figure 3.18. Backlash can be addressed in various ways, as discussed prior. Considering the shrinkage of parts inherent to the FDM process used to fabricate the gears, it was speculated that solving backlash in the gears is not the best way to go. The shrinkage means that a number of iterative changes to the exact gear shape would likely have to be made. This would have a negative effect on the adoption of the prototype by a wider audience, as repeated fine-tuning and reprinting are likely to cause frustration among potential early adopters, harming the image of this solution as accessible. Thus for the final iteration a belt driven variant was designed, described in the following section.

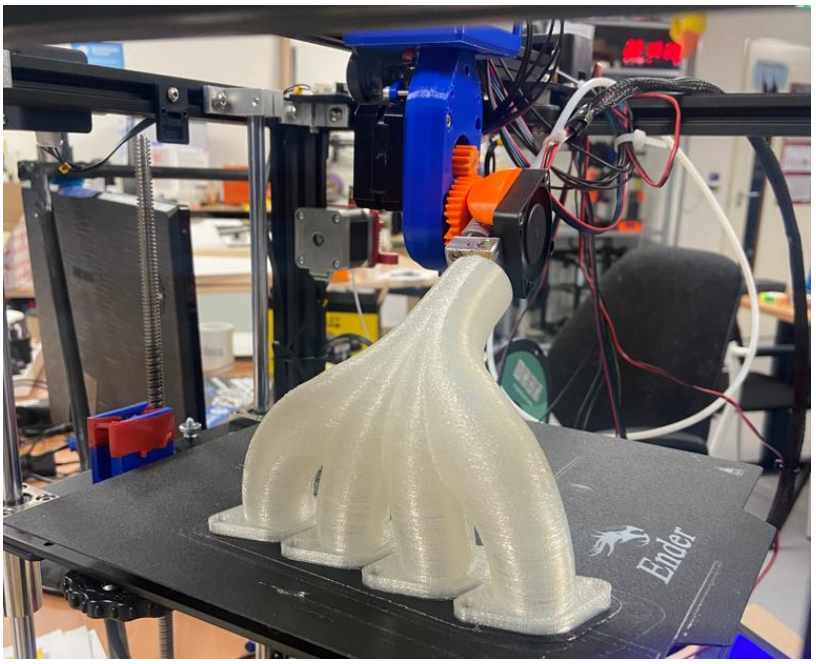


Figure 3.17: Manifold printed on RW1, sliced with G-code bending, illustrating capability to print large parts

The homing mechanisms of the rotational axes were not that reliable. This was due to the difficulty in fine-tuning their position. In this design, the engagement distance for the limit switches was adjusted by slightly shifting the switches position relative to the homing indents. To be adjustable like that, the switches were friction-fit, which loosened over time, and was overall not an elegant solution at all.

3.4 FINAL ITERATION

The final iteration of the print-head, RW2, builds on the RW1 design, addressing the two main issues - B-axis backlash, and homing reliability.

3.4.1 B-axis drive

To eliminate the backlash previously present in the B-axis gear drive, a belt drive was used instead. Belt drives have inherently less backlash, because, as the belt wraps around a section of the pulley's arc, it engages with numerous teeth. Belt drives are common in FDM printers, so the hardware is already familiar to users. The GT2 series of belt drive was used, because it is common in FDM, and as such could easily be acquired from the same stores that sell printer gear. A 20 tooth driving pulley was used, which is common, so it can even be harvested from a defunct printer. To find the right size suitable for the driven pulley the potential belt lengths were considered. The closed-loop belts come in certain lengths, so the sizes of the pulleys, the distance between them, and the belt loop length are all related. A simple 2-dimensional sketch was made to represent the rough layout alongside the known sizes of the driving motor and B-axis bearing. A timing pulley

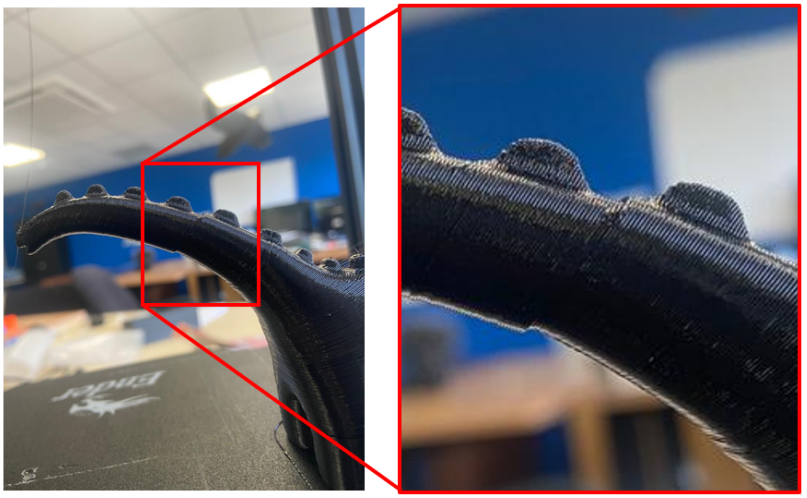
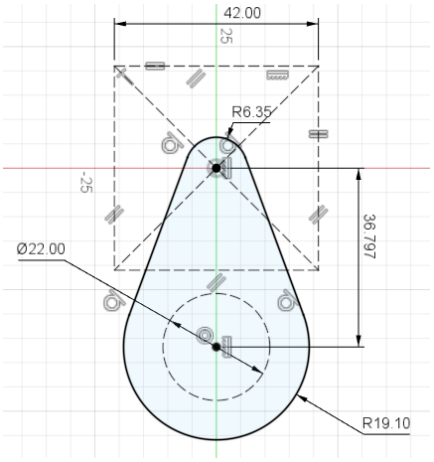
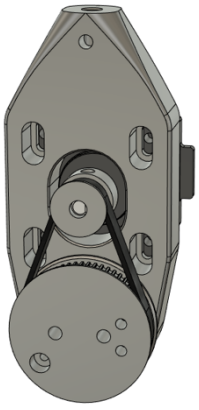


Figure 3.18: Layer shift in dinosaur print at roughly B=-45°, caused by backlash in gear drive

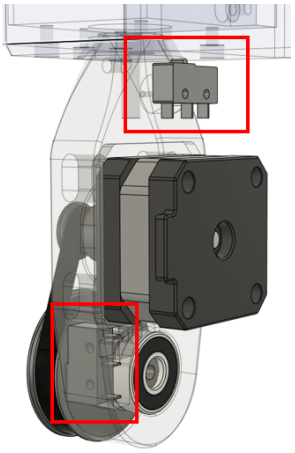
distance calculator tool was then used to iteratively arrive at a pulley size and belt loop length that fits. The layout sketch could then be fully constrained to represent the final layout, shown in Figure 3.19a. The 60 tooth driven pulley coincidentally also results in the same 3:1 reduction that was present in the B-axis of the RW1 prototype. The new "B-arm", shown in Figure 3.19b was then designed around the layout. The stepper motor is mounted in slots, so that the belt tension can be controlled. If the belt is not tense, the slack will cause backlash in a belt-drive.



(a) Layout of the B-arm



(b) Model of the B-arm, including pulleys



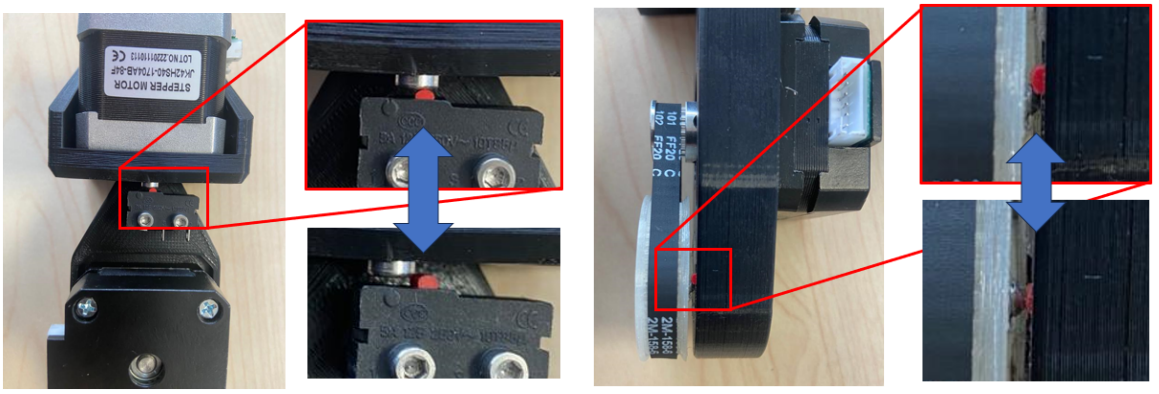
(c) Locations of limit switches for homing

Figure 3.19: RW2 B-arm design

3.4.2 Homing

The previously used principle of integrating limit-switches into the B-arm, which then interface with indents at the limits of the rotational axes was working in principle. Issues stemmed from a poor implementation, where the necessary adjustment of the engagement distance was inconve-

nient and not thought-out, requiring the precise shifting of the switches. For the final iteration, the switches were designed to be mounted to fixed positions on the B-arm, shown in Figure 3.19c. Instead, the feature that the switches engage with would be adjusted. Thus, instead of that feature being an indent designed into the opposite part, it is now a bolt, as shown in Figure 3.20. By tightening or loosening this bolt, the amount that it protrudes can be finely adjusted, which is used to precisely tune the engagement distance with the micro-switch.



(a) A-axis limit switch engaging with bolt head on X-axis gantry

(b) B-axis limit switch engaging with bolt tip on driven pulley

Figure 3.20: Limit switches for homing of RW2

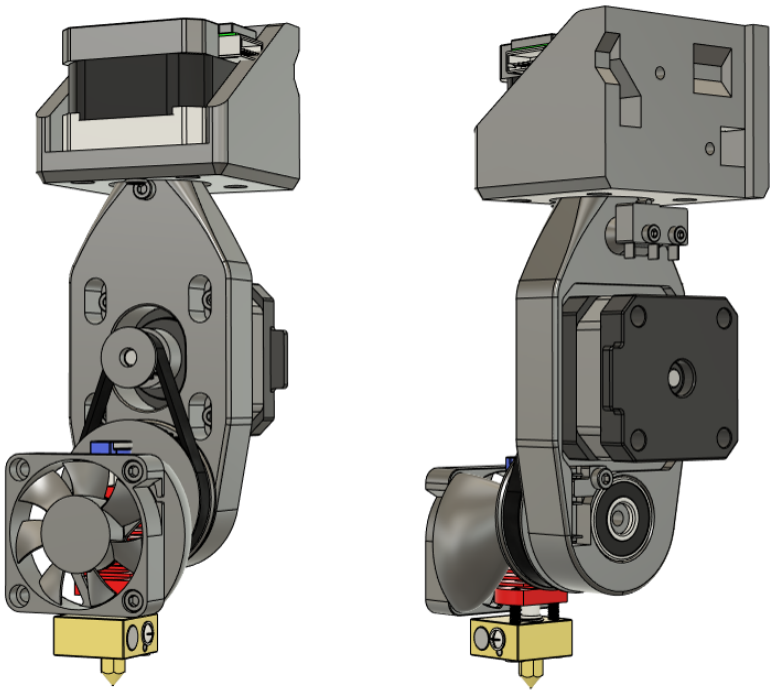


Figure 3.21: CAD model of RW2 print-head

Table 3.3 BOM of RW2 print-head

Item	*Qty*	*Part Number*	*Type*
1	1	Nema17 pancake stepper motor	COTS
2	1	GT2 20 T 5mm bore	COTS
3	1	B-driven-pulley	Print
4	*1*	*Hot-end-assem*	
5	2	608 2RS bearing DIN625	COTS
6	1	GT2 timing belt 6x158mm	COTS
7	1	B_arm	Print
8	1	Nema17 stepper motor	COTS
9	1	Carriage-mount	Print
10	2	Microswitch	COTS
11	1	B_switch_cover	Print
12	1	Fan-mount	Print
13	*1*	*Hot-end-fan*	
14	8	DIN912 M3x6mm sockethead bolt	Fastener
15	8	DIN912 M3x10mm sockethead bolt	Fastener
16	3	M3 heat-set insert	Fastener
17	1	Hotend_spacer	Print
18	2	DIN912 M2.5x10mm sockethead bolt	Fastener

3.4.3 Evaluation

This iteration proved accurate and reliable, as the B-axis backlash was eliminated, and the print-head homed reliably through extensive testing. The complete CAD model is shown in Figure 3.21. Like the previous prototypes, the design is simple and compact, with all the printable parts easily fitting within a single typical build-plate, as shown in Figure A.2. The bill of materials is given in Table 3.3.[11] The cost of the print-head, including the cost of COTS parts as sources for the project, as well as estimated print cost based on the price of filament[12] is €33, with a detailed breakdown given in Appendix A.1.

3.5 HOST PRINTER INTEGRATION

The print-head integrates with existing printers, between the X-axis gantry and the hot-end. The print-head mounts directly to the X-axis gantry. In this particular case the host printer is a Creality Ender 5 Pro, which has an X-gantry made of steel sheet metal, with various mounting holes and features. Two of these holes, normally used for mounting the hot-end shroud, are used to attach A-axis mount. One side of the mount has a flat locating feature, which butts up against the left edge of the Ender's gantry, to ensure the print-head mounts straight. For integrating into other printers the A-axis mount would have to be adjusted to fit whatever mounting options are present on that printer's gantry. There are few considerations or criteria for this, as long as the attachment is secure. If no locating feature can be integrated, more care must be taken when mounting.

[11] Italicized items are harvested from the host printer.
[12] Filament cost of €22.95 per 1 kg was used

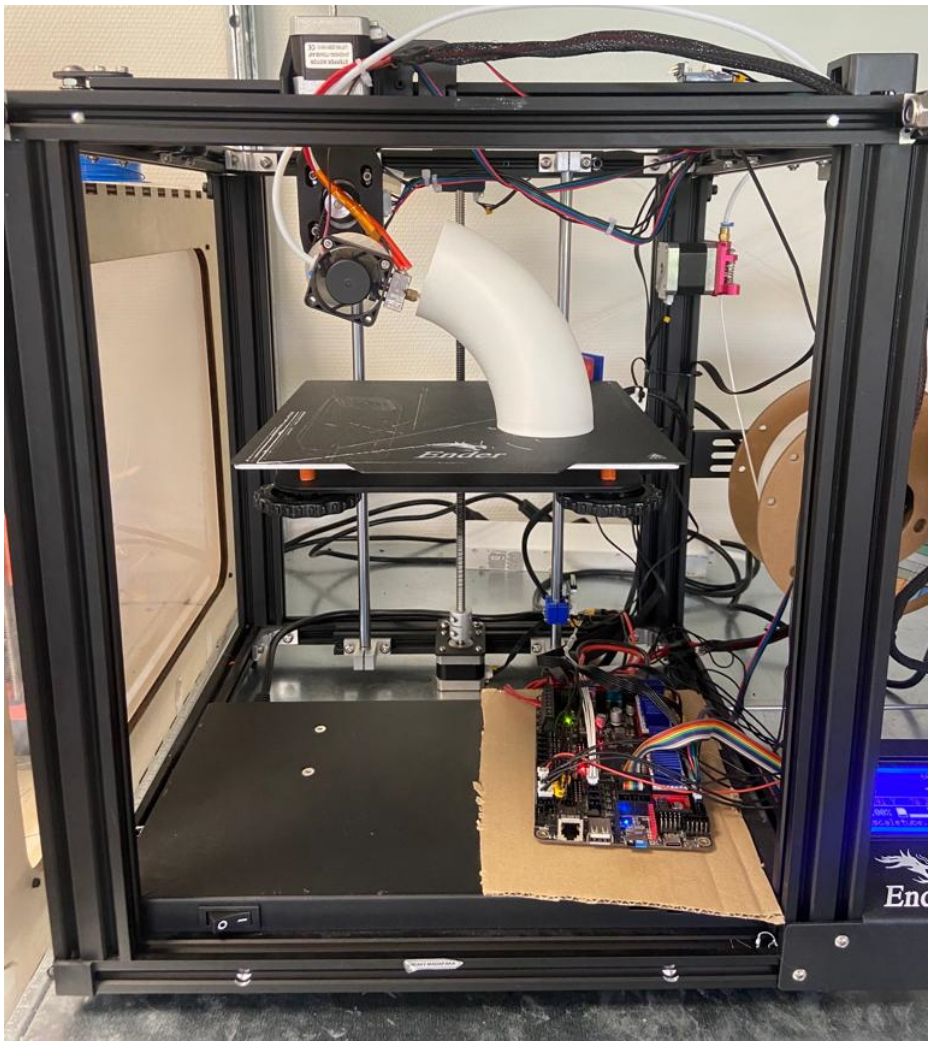


Figure 3.22: Ender 5 Pro with retrofit RW2 print-head, executing a large print

Additionally, for the RW prototypes the Z-axis limit switch must be moved from its Z-min position to a Z-max position. This is done by mounting the switch to one of the Z-axis linear rods using a 3D-printed adapter. A 3D-printed flange is used on the build-plate's corresponding linear bearing, as shown in Figure A.3. A full view of the host printer with the retrofit print-head is shown in Figure 3.22.

3.5.1 Electronics

While not at the center of attention, electronics are integral to ensure the functioning of any CNC machine, including our printer. The electronics of a FDM printer include motors and motor controllers that will drive the kinematic system and extruder, limit switches for all axes, heating elements and thermistors for the hot-end and printbed, a screen and input device for the user interaction,[13] a power supply, and a board that controls everything.

[13] A dedicated input and monitoring device is not strictly necessary, as the printer control boards can interface with personal computers.

Desktop FDM printers typically use stepper motors, as they provide a high accuracy while keeping low cost and simple control.[14] Stepper motors move in discrete steps, so they can be precisely controlled to achieve the precise motion necessary for the operation of a FDM printer. The Nema17 size has become the de facto standard stepper motor size for desktop FDM printers, and thus that same motor size will be used for the additional axes, which will ensure that performing the retrofit is as easy as possible and can be executed with spare parts.

The control board of FDM printers houses the microcontroller that runs the Firmware, and acts as a hub for all other electronics to connect. Controllers for the stepper motors (called stepper drivers), are typically housed on the control board - either integrated into the board or in sockets. Ideally the control board of the host printer could still be used for the 5-axis retrofit, however the boards typically support operation of up to 5 motors - X, Y, Z1, Z2, and E.[15] The required axes for the 5-axis retrofit are X, Y, Z, A, B, and E - a total of 6. The control board used for the project was the the BTT Octopus, populated with 6 TMC2208 stepper motor drivers, shown in Figure A.9, at a total electronics cost of €107.50.

[14] This is is contrast to higher-end printers that will use servo motors.

[15] The Z-axis is allotted two motor outputs, as setups with a dual Z-axis lead screw are common to distribute the load of large build-plates or X-axis gantries.

Chapter 4

Software design

CONTROLLING THE 5-AXIS HARDWARE calls for a specialized software stack that can adequately translate a 3D model into precise instructions for the printer's mechanical components. To execute a FDM print, a 3D model of the printable part must be produced using Computer Aided Design (CAD) software. The part geometry must then be converted into Geometric code (G-code), that the 3D printer can interpret. That process, called *slicing*, involves the use of algorithms to produce a path that the printhead will follow in order to produce the desired part. This G-code must then be translated into specific actions for the 3D printer's motors and heaters to perform, which is done by a Firmware running on the printer's control board. Figure 4.1 shows this generic software stack. At its core, the purpose of the software stack is to translate the design intent to the actions of the printer.

Figure 4.1: Generic FDM software stack

Multi-axis AM adds complexity to the software stack, due to the addition of DoF's. The path-planning becomes more involved in order to make use of the additional axes, as described in Section 2.3. And the firmware must now control and coordinate additional movements. The risk of interference between the printhead and workpiece or build-plate warrants some form of collision prediction or prevention, which may be integrated into the slicing environment. The introduction of rotational axes to the printer kinematics warrants the use of Inverse Kinematics (IK), as the linear axes no longer directly correspond to the cartesian coordinates requested by the slicer. This is handled by a set of mathematical expressions that translate the requested End Effector (EE) position to the required axis positions, and can be implemented either inside of the Firmware, or as a *translation layer* ahead of it. Considering these added requirements of collision prediction and IK, an expanded software stack for multi-axis FDM is shown in Figure 4.2.

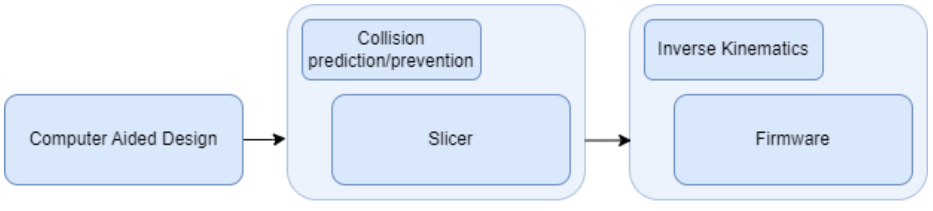


Figure 4.2: Multi-axis FDM software stack

4.1 PATH-PLANNING PRINCIPLES

This section explains the working principles behind path-planning for FDM, and what is added when introducing additional axes.

4.1.1 Translation

The first step to translating the CAD model into the printer actions that will manifest the desired part is path-planning, also known as slicing. It is the process of turning the digital data about the part geometry into G-code data, that the 3D printer can interpret, which is the most widely used programming language for CNC machines. Using it, we can send commands to the printer to execute certain actions, such as moving motors to a set position at a set speed, or raising the temperature of the nozzle or printbed. I.e. we tell the printer what to do. This simple functionality is in principle all that is needed to execute a successful printjob.

Figure 4.3: Deposition of a single track of material by giving a G-code command.

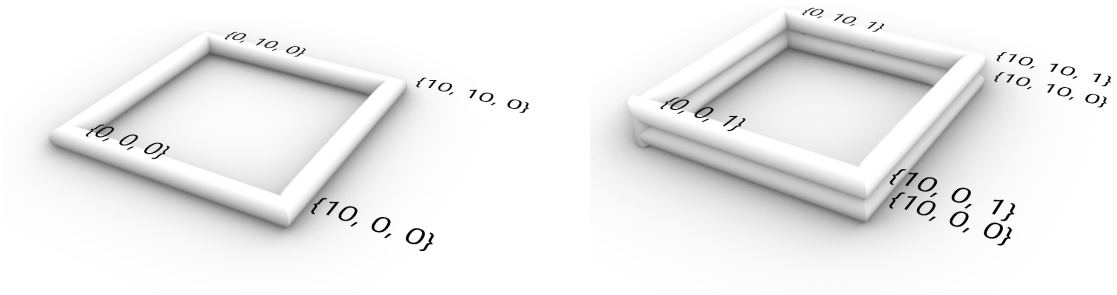
The hot-end of a FDM printer deposits molten thermoplastic material through the orifice of a nozzle. By extruding material while also moving the printhead we can deposit a track of material, as shown in Figure 4.3. This command is given using G-code. Each line of G-code constitutes one command, the most common of which is the "linear movement" command, or G1. G1 commands the printer to linearly move from its current position to the commanded position.[1] For example, the following command line

```
G1 X10 Y0
```

would make the X-axis move 10 units to the positive side. Additional axes can be simply appended to the command. E.g. to produce the singular track of material shown in Figure 4.3, the command could be

```
G1 X10 Y0 E10
```

[1] If the printer is in absolute coordinate mode (G90). In relative coordinate mode (G91) the printer will move the relative distance from its current position.



(a) Singular contour defined using X and Y coordinates

(b) 3-dimensional shape defined using X, Y and Z coordinates

Figure 4.4: Printing a contour using a sequence of G-code commands.

which would move the X-axis while extruding material.[2] Lines of G-code can be stringed together to produce a sequence of movements. By ordering the movement to each of the corners of a square while extruding another 10 units for every edge, we would print the square shown in Figure 4.4a using the following G-code:

```
G1 X0 Y0 E0
G1 X50 Y0 E10
G1 X10 Y10 E20
G1 X0 Y10 E30
G1 X0 Y0 E40
```

Several layers of this square can be built on top of one another by increasing the Z-height between iteration of the shape, resulting in the follow 3-dimensional shape show in Figure 4.4b, using the following G-code:

```
G1 X0 Y0 Z0 E0
G1 X50 Y0 Z0 E10
G1 X50 Y10 Z0 E20
G1 X0 Y10 Z0 E30
G1 X0 Y0 Z0 E40
G1 X0 Y0 Z1 E40
G1 X50 Y0 Z1 E50
G1 X10 Y10 Z1 E60
G1 X0 Y10 Z1 E70
G1 X0 Y0 Z1 E80
```

Dividing any polyline into points allows us to print an approximation of any shape, as shown in Figure 4.5. The accuracy of the representation depends on the amount of divisions that are used. The basic path-planning algorithm for typical FDM involves slicing the part geometry into equally spaced layers (hence the term *slicing*), parallel to the build-plate, each of which is a 2-dimensional shape. The perimeter of this shape is then divided into points in the same fashion as

[2] As explained in Section 3.5.1, the E axis is the extruder motor, which pushes filament of material into the hot-end.

(a) Contour with rough division of polyline

(b) Contour with fine division of polyline

Figure 4.5: Division of polyline into points for giving linear move G-code commands

shown previously. Whereas the internal area of the shape is filled with an infill geometry. This process is repeated for each layer to build the entire part.

4.1.2 Rotation

When slicing for additional rotational axes we can define the angle of the axes at each point. In the case of the 5-axis printer, we have angular axes A and B. A vector can be denoted for each point, which corresponds to the direction that the nozzle should be facing at that point. The nozzle-vector[3] can then be deconstructed into A and B coordinates. The A coordinate is the angle between the vector's projection on the XY plane and an arbitrary reference axis, e.g. X or Y. The B coordinate is the angle between the nozzle-vector and the Z axis. An example track with nozzle-vectors is shown in Figure 4.6, and would be achieved using the following G-code commands:

```
G1 X0 Y0 Z0 E0 A0 B0
G1 X10 Y0 Z0 E10 A0 B45
```

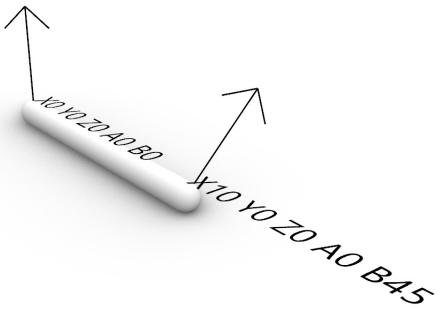


Figure 4.6: Deposition of a single track of material with nozzle direction defined using vectors.

[3] In CNC terms often called "tool-vector".

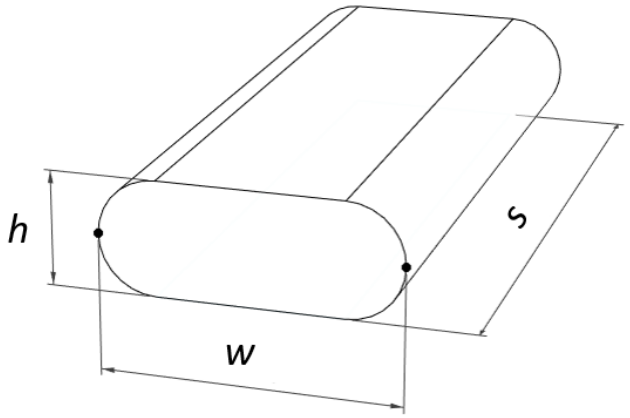


Figure 4.7: Parameters of a deposited track, used for flow calculation

4.1.3 Extrusion

The E value in a `G1` command requests the amount of material extruded. This value represents units of length (mm) of filament extruded. So, how do we know how much length of filament must be extruded to deposit the desired track? Because whatever material goes into the hot-end, must also come out, there is an equivalence between the volume of the deposition and the volume of the extruded filament. In both cases, we can express the volume as the product of the cross-section of the material and the length. On the filament-side, the cross-section is circular. Whereas on the deposition-side the approximation of the track's cross-section is a rectangle with semi-circular sides, pill shaped, as shown in Figure 4.7. We can formulate the volume equivalence as shown in Equation 4.1, where E is the flow or extrusion rate, h is the layer height, w is the track width, s is the track length, and d is the filament diameter. Flow (E) can then be expressed as shown in Equation 4.2. Somewhat unintuitively, the nozzle size is not part of the equation that defines flow, instead it sets the range of possible layer heights.

$$\pi \cdot E \cdot \left(\frac{d}{2}\right)^2 = \left((w-h)\cdot h + \pi \cdot \left(\frac{h}{2}\right)^2\right) \cdot s \tag{4.1}$$

$$E = \frac{\left((w-h)\cdot h + \pi \cdot \left(\frac{h}{2}\right)^2\right) \cdot s}{\pi \cdot \left(\frac{d}{2}\right)^2} \tag{4.2}$$

4.1.4 Machine routines

The G1 linear move commands are the bulk of the G-code file, commanding the movement required to execute the print-job. But it sits sandwiched between what is called the start and end G-code. These machine routines are a set of commands that the printers perform at the beginning of each print to initialize, and at the end, to ensure a proper termination of the print job. There are

certain actions that the start G-code will typically command. Firstly, before any movements, the printer will be instructed to perform a homing movement using the G28 command. Coordinates will be set to absolute using G90. The temperatures of the nozzle and print-bed will be set, and the printer will wait for the defined values to be reached. Once up to temperature, the printer will usually print a prime line, to stabilize the flow, before beginning the print. End G-code is generally simpler, only moving the hot-end out of the way and powering down the motors. The machine routines are typically specific to the printer, and come loaded with the printer profiles of slicers, where they can also be edited by the user. For the purposes of the multi-axis printing, there are a few changes made to the start routine. Normally the homing is commanded once the nozzle is already part-way heated up, as a time-saving measure. However, the specific homing position had on one occasion caused part of the hot-end to come into contact with some wires, posing a melting hazard. Therefore, the homing is commanded before any heating. Additionally, once the rotational axes are homed to their minimum positions, they would normally stay there. Instead the `G1 A0 B0` command is given immediately after `G28` to return the rotational axes to their `0` position instead of the minimum position. The start and end routines that were modified and used for the GH scripts were from the Ender 5 Pro profile in Prusa Slicer. The full routines are shown in Appendix C.1.

4.2 ISO-CURVES SLICING

Iso-curves slicing utilizes the properties of isoparametric curves to divide a parametric surface into an infinite amount of lines that constitute that surface, as explained in Section 2.3.2.1. The curves are given by a function $f(u, v)$, for which one parameter (u or v) is fixed.[78] The use of iso-curves for non-planar slicing is beneficial, because it is an automatic way to define layers that naturally follow the shape of a surface, as shown in Figure 4.8.

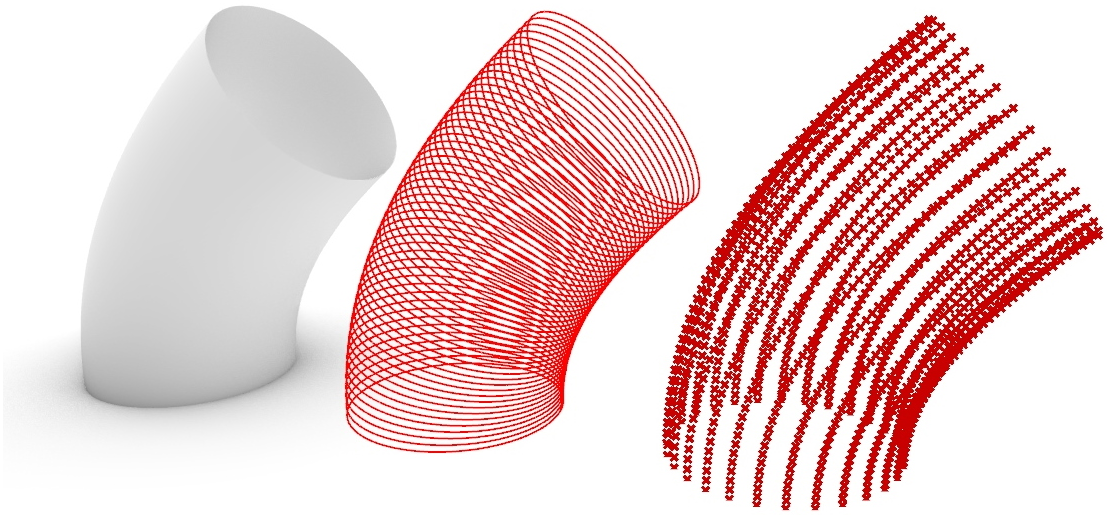


Figure 4.8: Example of iso-curves generated for simple part. Surface, iso-curve layers, points.

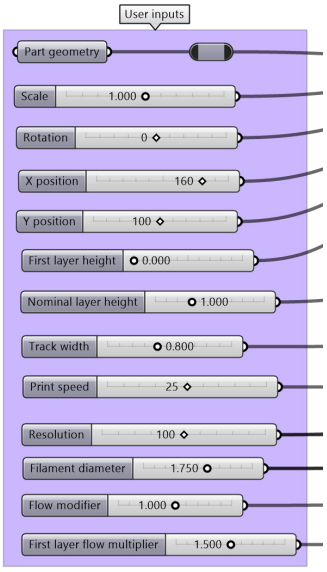


Figure 4.9: User inputs for script

The implementation of the iso-curves slicing in GH uses the previously described heuristic of dividing a geometry into a path or curve that the printer would follow. Then dividing that path into X, Y and Z points. Then defining a nozzle vector for each point, and extracting its angles. Flow is calculated using the equation explained prior. However, many "modules" are required to make the complete slicing script. They will be explained step-by-step. Some of the script is based on the principles described by Cuevas.[33] The script is compartmentalized into clusters, which are groups of components that are combined to function as a single component within GH. The script is shown in segments that are small enough to be understandable, as the full script is too extensive to show visually. The contents of the clusters being referred to can be found in Appendix B.1, as well as a link to the GitHub repository containing the full GH file.

4.2.1 User inputs

The first module of the script is the user inputs, which is where the user defines the geometry to slice, alongside the slicing parameters. Parameters are manipulated using sliders, as shown in Figure 4.9, which have a default value, and also limits that relate to the reasonable range of values expected for that parameter.

Positioning and scaling The first set of parameters are used to position and scale the input geometry. While this could just as well be done in the Rhinoceros interface, it is preferable that the input geometry is simply imported into Rhino into its default positions, and any further manipulation is done through the GH script. This workflow makes it easier to recreate scenarios, as everything is defined parametrically. The parameters available here are: Scale, rotation, X position, and Y position.

Layer height The layer heights that can be defined are the *first layer height* and *nominal layer height*. The first layer height can be independently adjusted, as is typical for slicers, as this may be best to tune for specific build-plate surfaces. The nominal layer height is the user-set approximate layer height for the rest of the part, since the layer height is variable.

Track width The track width defines the physical width of the extruded track in mm. A wide range of track widths are possible for FDM, depending on the nozzle size and specific nozzle geometry.[95]

Print speed The user-set print speed (in mm/s) commands the feed-rate of the printer, but is overridden by the limits defined in the printer's firmware or the start G-code routine.

Resolution The resolution is the amount of points that each layer is divided into, and is the primary determining factor of the extent of tessellation that will be present in the print (shown earlier in Figure 4.5). Increasing the resolution requires additional computation resource, as the number of computable points increases.

Filament diameter The filament diameter must be defined, as it is used for the flow calculation. Even though it is generally always going to be 1.75 mm for the typical consumer FDM printers, the script still allows for this to be changed in case a different extruder is used.

Flow The final parameters are modifiers for the flow. They are split into a general flow modifier, and one for just the first layer. The *first layer flow modifier* is set at 1.5 by default, as it is beneficial to overextrude the first layer to ensure good adhesion and flow stabilization.[4] The general *flow modifier* is commonly present in slicers as an easy way to fine-tune the extrusion rate, in case there is slight over- or underextrusion.

4.2.2 Division into points

A sequence of actions, shown in 4.10, are taken to arrive at the point when the input geometry has been divided into points, as shown in Figure 4.8. First, the geometry is positioned using the *positioning* cluster, which is shown in Figure B.3. The positioned geometry is then fed into the *Iso Curve* component of GH. It also requires points at which to place the iso-curves. Thus the amount of these equally spaced points determines the amount of layers. The points are generated in the *Layer count* cluster, by dividing the Z-size of the geometry's bounding box by the user-set *nominal layer height*, as shown in Figure B.4. The V-isocurves are then divided into points, using the user-set resolution as the point count. This gives us points that would be sufficient to execute the XYZ motion required to print the geometry, shown in Figure 4.8.

[4]When first starting the print, the flow can be a bit unpredictable, as some may have oozed out during warmup. Flow stabilization is simply the idea of extruding additional material ahead of the bulk print to ensure that there will be material to extrude.

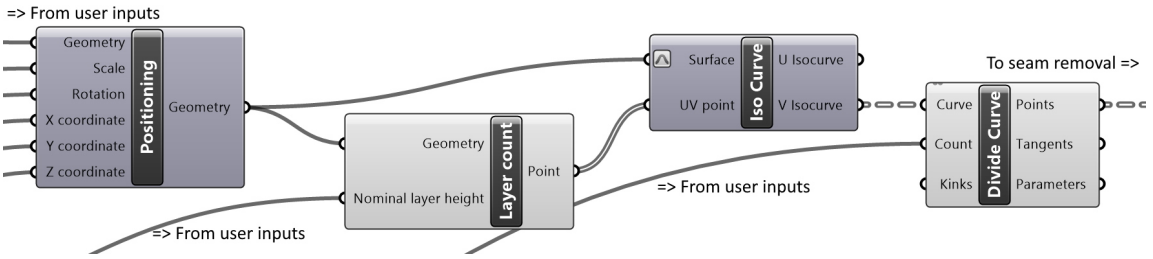


Figure 4.10: GH definition to go from user inputs to XYZ points

4.2.3 Nozzle vectors

Once we have the points for XYZ motion, we are still missing information about rotation - A and B. The continuation of the script is shown in Figure 4.11. The sequence of components that execute the vector discovery are within the *Nozzle vectors* cluster, shown in Figure B.5. First the matrix of the XYZ points is flipped, and the vertices are used to generate polylines. Flipping the matrix swaps the rows and columns of the data tree, so the points now follow perpendicular iso-curves, as shown in Figure 4.12a. The polyline is then exploded into segments, each segment representing the polyline between any two adjacent points, as shown in Figure 4.12b. The start and end point of these segments are then used to generate the vectors, as shown in Figure 4.12c.[5] The A and B angles are then extracted from the vectors with the *Vector angle extraction* cluster, shown in Figure B.6.

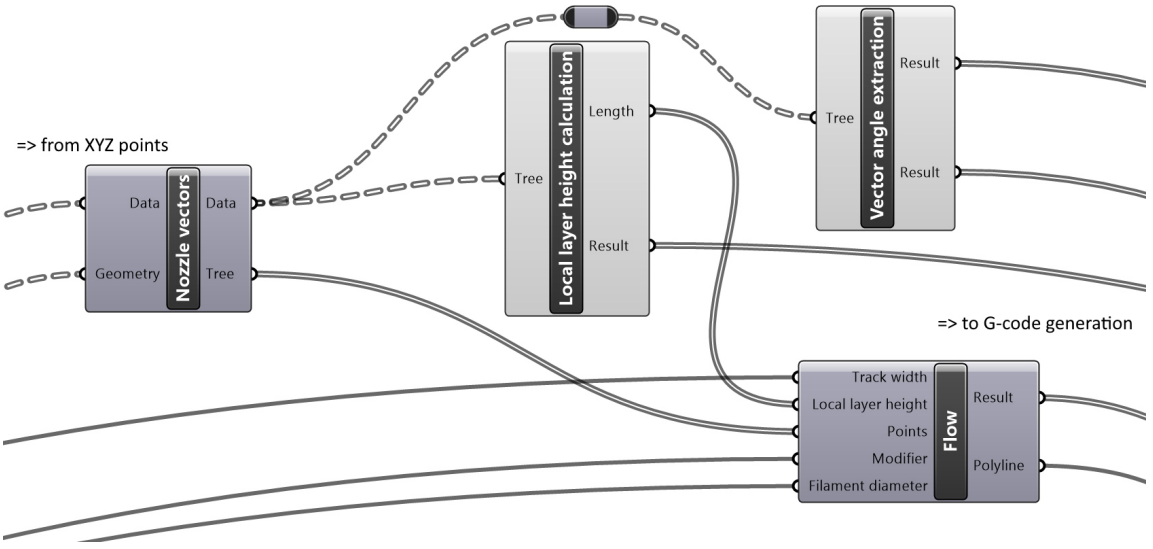


Figure 4.11: GH definition to go from XYZ points to nozzle vectors

[5]The vectors are also used for seam-removal.

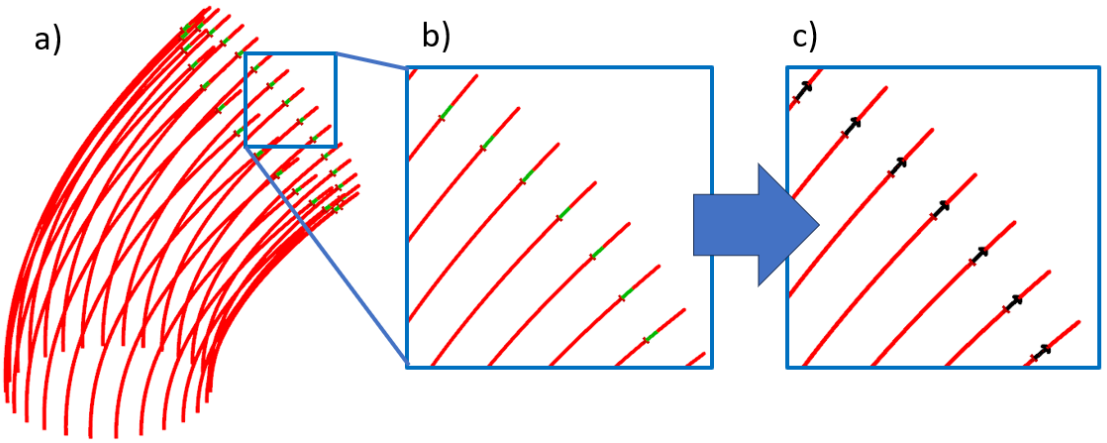


Figure 4.12: Vector discovery process through flipping point matrix: (a) Polylines of flipped matrix; (b) Exploded segments of polylines; (c) Vectors formed from start- and end-points of exploded segments

4.2.4 Flow

The only remaining piece of the puzzle is the E value - flow. Calculating the flow is trivial, despite the non-planar nature of the toolpaths rendering the layer height is variable, making it more complex than for planar slicing. Equation 4.2 described in Section 4.1.3 is applied, which relates the volume of the desired track to the volume of the extruded filament, as both have to be equal. The equation uses the track width, nozzle diameter and segment length to calculate the flow. The user-set flow multiplier is also applied, as shown in Figure B.7. With all the necessary values, X, Y, Z, A, B, and E, all that remains is to concatenate them into strings of G-code,[6] and adding the start- and end-routines described in Section 4.1.4, shown in Appendix C.1.

4.3 G-CODE BENDING

G-code bending is a form of G-code anamorphosis method, in which a morphed version of a part is sliced using a typical planar slicer, following which the toolpaths are morphed the opposite way to restore the object's shape, as described in Section 2.3.3. This method, developed by Hermann, uses a Python script to bend toolpaths to the shape of a spline.[81] It was initially developed as a way to improve print quality of slightly curved objects using a typical 3-axis FDM printer equipped with a high-clearance hot-end. As the method bends toolpaths to a spline, it remains limited to use for objects with a bent longitudinal axis.

A core benefit of G-code bending, and G-code anamorphosis in general, is that the refined functionality of FDM slicers is used as the basis. When devising the previous slicing algorithm from scratch in GH it became clear how many nuances there are in the common slicing algorithms. Such as deconstructing the part into external and internal perimeters, top and bottom top and internal infill, bridges, skirts and brims. Each of which has a different way of generating the paths.

[6] After applying the IK translation, explained later.

Between each of these features there are travels, which include mechanisms such as retraction of the filament material, wipes, and z-hop.[7] By using a G-code anamorphosis approach, more than a decade worth of refining the open-source slicing algorithm is maintained, rather than having to be remade.

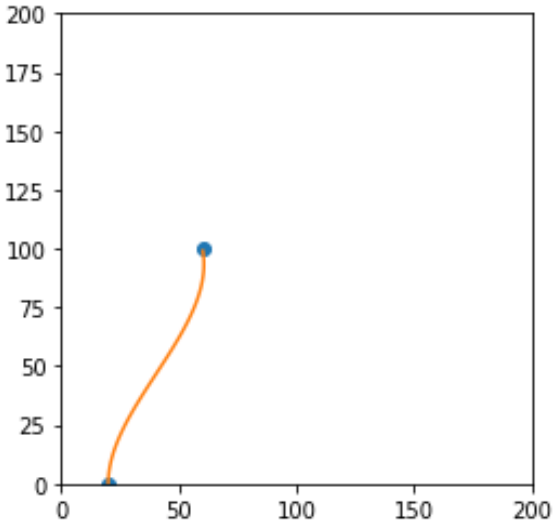


Figure 4.13: Default spline for G-code bending by Hermann, generated by Python script[82]

In the original script by Hermann, the user defines as spline using the `CubicSpline` function of the SciPy library for Python. A number of points on the XZ plane can be defined for the spline to pass through, alongside boundary conditions determining the angle of the spline at either end, as shown below. The script also generates a graphical representation of the spline, as shown in Figure 4.13.

```
SPLINE_X = [40, 140]
SPLINE_Z = [0, 115]
SPLINE = CubicSpline(SPLINE_Z, SPLINE_X, bc_type=((1, 0), (1, -np.pi/6)))
```

For each layer, the midpoint of the layer is found using

```
midpointX = lastPosition.x + (currentPosition.x - lastPosition.x) / 2
```

Then the distance from that midpoint is found using

```
distToSpline = midpointX - SPLINE_X[0]
```

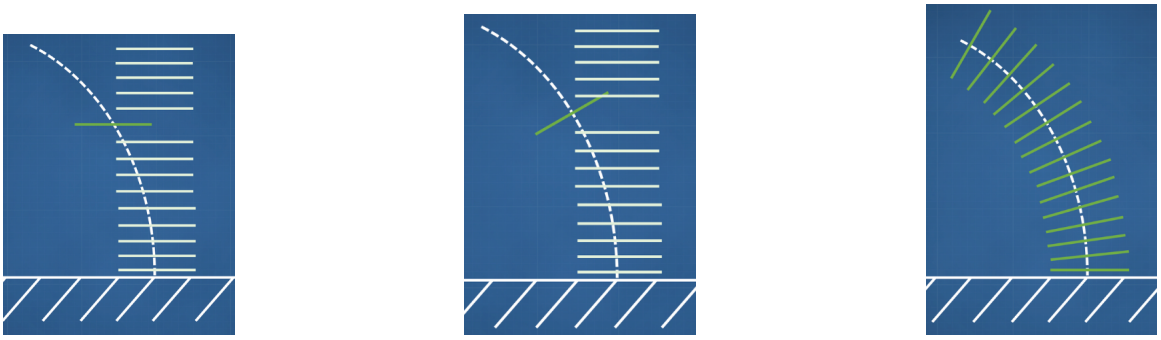
The angle of the spline at that layer is found using

```
angleSplineThisLayer = np.arctan(SPLINE(correctedZHeight, 1))
```

[7] These are movements designed to ensure the clean and defect-free movement between different features of the print.

The toolpath is then transformed, by translating it by the distance to the spline `distToSpline` and rotating it by the angle of the spline as follows:

```
transformedGCode = getNormalPoint(Point2D(correctedZHeight,
    SPLINE(correctedZHeight)), SPLINE(correctedZHeight, 1),
    currentPosition.x - SPLINE_X[0])
```



(a) Path is translated (b) Path is rotated (c) The process is repeated for all layers

Figure 4.14: Working principles of G-code bending by Hermann[81]

Due to the non-parallelism of layers, the distance between them is shorter closer to the center of the rotation arc, and vice versa - the layer height is variable. To prevent overextrusion where the local layer height is smaller, and underextrusion where it is larger, the flow must also be variable. In G-code bending, the flow is calculated during the initial slicing, so the extrusion values are then corrected during bending, using the following function. It is, however, a simplification of the true behavior, as it assumes a linearly proportional relationship.[8]

```
extrusionAmount = float(currentLineCommands.e) * ...
((LAYER_HEIGHT + heightDifference)/LAYER_HEIGHT)
```

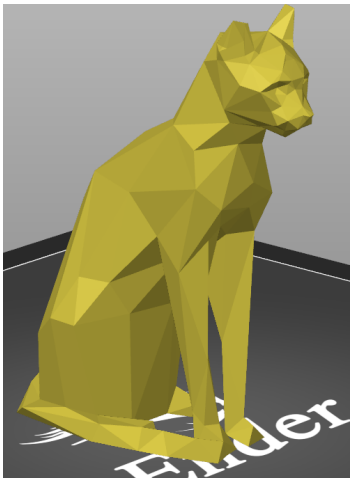
To adapt the technique for multi-axis, the original script is modified. For each layer, the angle of that layer must be fetched and included into the `writeLine` command, which writes the recalculated coordinates, as shown below. The full modified script is shown in Appendix C.3.

```
def writeLine(G, X, Y, Z, A, F = None, E = None):
outputSting = "G" + str(int(G)) + " X" + str(round(X,5)) + " Y" + ...
str(round(Y,5)) + " Z" + str(round(Z,3)) + " A0 B" + str(round(A,3))
Baxis = angleSplineThisLayer * 57.2958
writeLine(1,transformedGCode.y, currentPosition.y, ...
transformedGCode.x, Baxis, None, extrusionAmount)
```

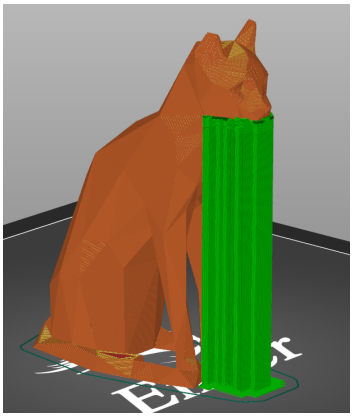
The original author displays a use case for this approach, where straight parts are designed in CAD, then the G-code bending is applied, meaning that the intended curvature of the final part

[8] Ellipsis (...) used for line-breaks in code.

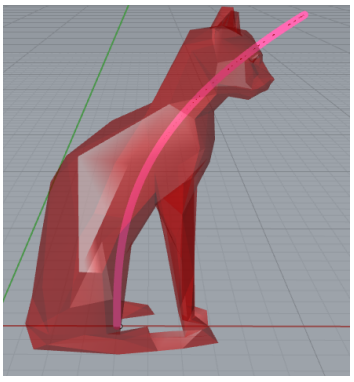
is not actually present in the CAD data. Thus, to apply this technique to general parts, the part geometry has to be bent the opposite way before slicing. As an example we can consider the low-poly model of a sitting cat, shown in Figure 4.15a. The forward-facing chin of the cat is an overhanging structure, and would require support-material, as shown in Figure 4.15b. It is possible to identify a longitudinal curve, which could be used for bending. Looking at Figure 4.15c you can see that the prior overhanging chin forms a roughly 45°angle with the spline. The ears, which are otherwise straight up, also form a roughly 45°angle with the spline. This means that the part can be straightened along this spline to slice it support-free.



(a) Sitting cat model



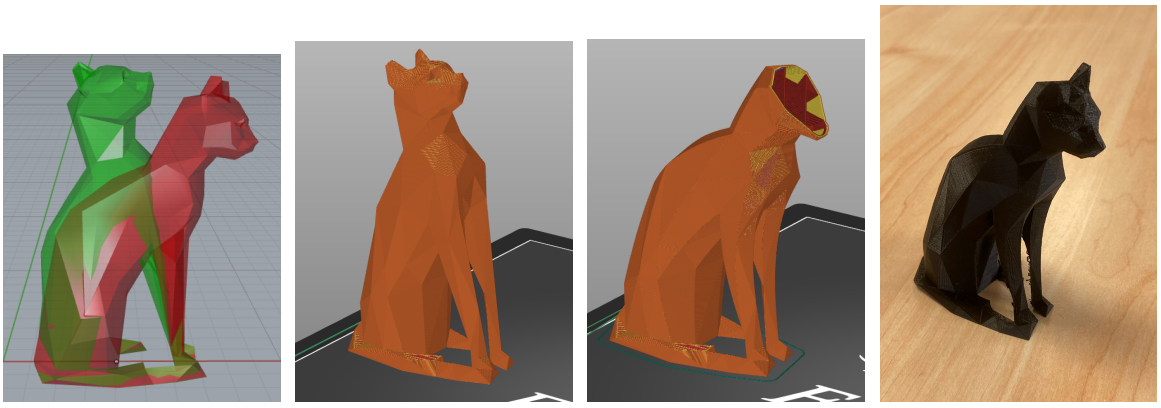
(b) Cat model sliced with Prusa slicer, illustrating extent of required support structures



(c) Spline that can be used for bending of cat geometry to limit overhang angles to roughly 45°

Figure 4.15: Example for part that normally requires support structures, and how a spline can be fitted to it for G-code bending

The first step is to prepare the part mesh for slicing. That is done in GH using the "Morph Flow" component. We define the part mesh as "Input geometry" and the previously denoted curve as the "Base". For "Target" we construct a straight line of the same length, which starts at the same point as the "Base". The output of the component is a mesh of the part geometry, which has been straightened, as shown in Figure 4.16a. This straightened cat model can then be exported, to be easily sliced without support structures, as shown in Figure 4.16b. The resulting G-code is then run through the bending script. The bent G-code can be viewed with a G-code viewer, commonly included with slicer software. Examining Figure 4.16c, we can see that the final paths now represent the initial state of the cat, and the layers are no longer parallel to the build-plate. A print using this G-code is shown in Figure 4.16d. Applying this workflow, the end result is not geometrically identical to the initial part, presumably as the morphing algorithm within GH works differently than that written by Hermann. However, this illustrates how, fundamentally, this technique can be effectively used to eliminate support structures for overhanging geometry of certain parts. Meanwhile, the specifics of the implementation can be improved to ensure retention of the exact part geometry through the anamorphism process. More parts sliced and printed using this technique are shown in a later chapter, in Figure 5.2.



(a) Initial model in red, morphed model in green. (b) Sliced morphed model (c) Bent G-code, showing cross-section (d) Executed print

Figure 4.16: Workflow for G-code bending of sitting cat model

4.4 FIRMWARE

Some of the preceding sections covered the process of generating 5-axis G-code. This G-code is read by the firmware on the printer's control board, and the given commands are executed. Due to the lack of 5-axis FDM printers, there are no off-the-shelf fully functional 5-axis FDM firmware. Three potential paths to solve this were considered - writing a firmware from scratch, adapting an existing 5-axis CNC firmware to support FDM printing, and adapting an existing FDM firmware to support 5-axis movement.

The option of writing the firmware from scratch was considered, as the fundamental workings of the firmware are simple. In order to execute the G-code commands, the firmware must simply read coordinates and move motors to match the new coordinates. Any complexity related to the movement is handled by the slicer during path-planning. Some auxiliary functions would also need to be implemented, such as responding to temperature commands, and a PID control loop for keeping constant nozzle and bed temperatures. The benefit of writing from scratch would have been a deep understanding of the underlying code base, which would make it trivially simple to implement IK. On the other hand, the end result would have been rough, as it would be impossible to achieve a polished solution considering the scope and timeframe of the project.

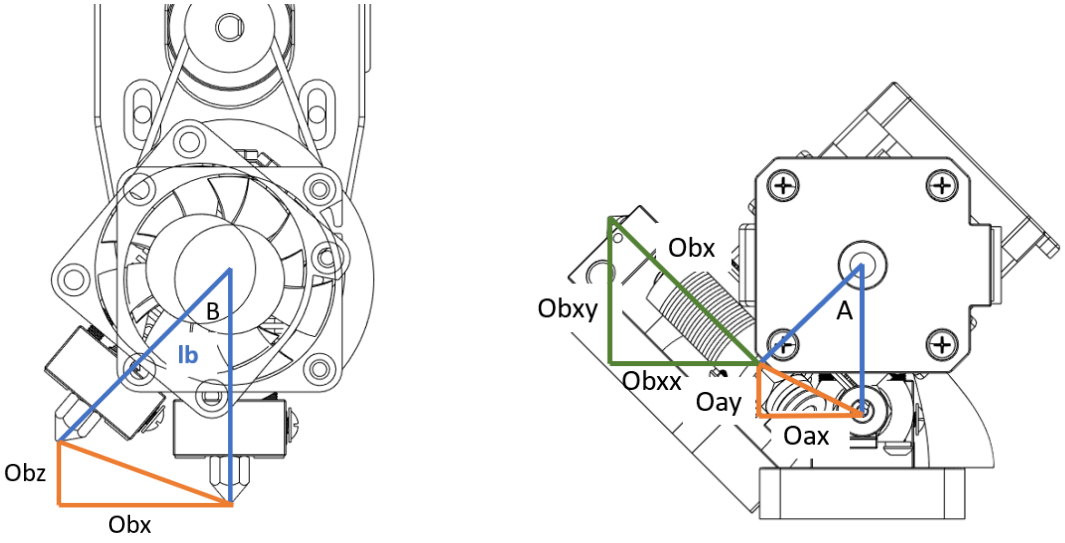
Adapting an existing 5-axis CNC firmware to support FDM printing features would have been a feasible route, as it already has all of the kinematics covered. Additions would have had to enable the firmware controlling the extruder and heating. Grbl is a popular open-source CNC firmware that runs on Arduino, thus there was reason to believe that executing this modification of grbl would be feasible.

Finally, adapting a FDM firmware to support additional axes would ensure that all of the printing related features are present, and only the kinematics of two additional axes need to be sorted out. Marlin is a common open-source FDM firmware by van der Zalm and Kubicek, and is originally based on Grbl and Sprinter firmwares.[96] Marlin already has inbuilt support for 6 additional axes of movement, 3 rotational and 3 linear, internally labelled I, J, K, U, V, and W, respectively. Since they are not typically used, they have to be configured in order to be used. To

find all the required parameters, the I and J axes were enabled in the `Configuration.h` file of the Marlin build, and then through iterative attempts to compile the firmware errors were identified and addressed. Parameters in the Configuration.h file that had to be configured include the homing direction (`i_HOME_DIR`)[9], endstop directions (`USE_iMIN_PLUG`), and homing speeds (`HOMING_FEEDRATE_MM_M`). In the advanced configuration (`Configuration_adv.h`) axis must be set to relative or absolute mode (`AXIS_RELATIVE_MODES`), as well as a homing bump[10] value must be set (`HOMING_BUMP`). Additionally, pins for for the stepper outputs and end switches had to be defined in the `/Marlin/src/pins/` directory, as they are not set by default.

4.5 INVERSE KINEMATICS

Forward Kinematics (FK) and Inverse Kinematics (IK) are expressions that relate the joint variables of a mechanism, to the position of the End Effector (EE). Forward Kinematics (FK) express where the EE will be given a set of joint variables. IK express what the joint variables have to be inorder for the EE to be at a set position. In the control of complex mechanisms, IK are required to translate the user-given positional requests into the actual joint parameters of the mechanism. The most common kinematic system for FDM printers does not require any IK at all, because the linear axes of the mechanism correspond to the axes of the Cartesian coordinate system, in which the position is requested. E.g., if the EE is requested to be at 10 mm from the origin, the X axis must simply move to the 10 mm position to fulfill that request. Other less common FDM kinematic structures, such as delta or polar, require IK.



(a) Front, XZ view

(b) Top, XY view

Figure 4.17: Links and geometric dimensions used in deriving IK solution.

[9] Lowercase i is axis name internal to Marlin (X, Y, Z, I, J)

[10] Also known as re-bump, the distance that an axis retracts after homing before the repeated homing. Necessary in case an axis is already triggering the end-switch.

4.5.1 Formulation

Considering the added rotational axes of the retrofit printhead, IK is required to calculate joint parameters whenever $A, B \neq 0$. Let's consider the B-axis motion. If the B-axis is moved, the tip of the nozzle has also moved in the X and Z directions, as shown in Figure 4.17a (looking at the XZ plane). These offsets caused by the B-axis motion in X and Z directions (X_{oB} and Z_{oB}) can be calculated using simple trigonometry, as shown in Equations 4.3 and 4.4, where the hypotenuse l_B is the distance between the B-axis and the nozzle's tip in the XZ plane.

$$O_{bx} = sin(B) \cdot l_B \tag{4.3}$$

$$O_{bz} = l_B - cos(B) \cdot l_B \tag{4.4}$$

Then let's consider the A-axis motion. Moving the A-axis also moves the nozzle's tip in the X and Y directions, as shown in Figure 4.17b. These offsets X_{oA} and Y_{oA} can then be expressed as before. The IK formulation is given in Equation 4.5, where the joint positions X, Y, Z, A, B are defined using the position and angle of the EE given in X', Y', Z', A', B'. The full derivation is shown in Appendix A.2. This is what is called a direct or closed-form solution, as the kinematic relationship can be expressed as a mathematical formula that has been derived analytically. More complicated mechanisms require an iterative numerical approach.

$$\begin{bmatrix} X \\ Y \\ Z \\ A \\ B \end{bmatrix} = \begin{bmatrix} X' + sin(A') \cdot l_a + cos(A') \cdot sin(B') \cdot l_b \\ Y' - l_a + cos(A') \cdot l_a - sin(A') \cdot sin(B') \cdot l_b \\ Z' + cos(B') \cdot l_b - l_b \\ A' \\ B' \end{bmatrix} \tag{4.5}$$

4.5.2 Translation layer

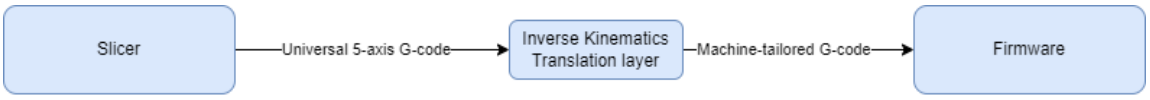


Figure 4.18: Schematic represenation of IK translation layer

As the public release of Marlin does not have provisions for the implementation of such custom closed-form IK,[11] instead a translation layer is used between the slicing and firmware. This is implemented as a Python script, which recalculates the axis values of a G-code file using the provided equations. As input, the script uses universally understood 5-axis G-code that would be processed by any 5-axis slicer, and outputs G-code tailored to the specific kinematic structure and link lengths of a specific machine, as shown in Figure 4.18. The script parses lines of G-code, ignoring M commands and storing the coordinate values in floats:

[11] Even though it has IK for certain pre-determined kinematic structures, like delta, polar and scara.

```
for i in range(1, len(parts)):
if parts[i].startswith("X"):
    x_val = float(parts[i][1:])
    x_indices.append(i)
elif parts[i].startswith("Y"):
    y_val = float(parts[i][1:])
    y_indices.append(i)
elif parts[i].startswith("Z"):
    z_val = float(parts[i][1:])
    z_indices.append(i)
elif parts[i].startswith("A"):
    a_val = float(parts[i][1:])
elif parts[i].startswith("B"):
    b_val = float(parts[i][1:])
```

Values for link lengths `la` and `lb` are defined. The coordinate values are then recalculated based on the IK solution. The script has functions for the recalculation of X, Y and Z values, whereas A and B are ignored as they remain unchanged:

```
def func_x(x_val, y_val, z_val, a_val, b_val):
    return x_val + math.sin(math.radians(a_val)) * La + ...
    math.cos(math.radians(a_val)) * math.sin(math.radians(b_val)) * Lb

def func_y(x_val, y_val, z_val, a_val, b_val):
    return y_val - La + math.cos(math.radians(a_val)) * La - ...
    math.sin(math.radians(a_val)) * math.sin(math.radians(b_val)) * Lb

def func_z(x_val, y_val, z_val, a_val, b_val):
    return z_val + math.cos(math.radians(b_val)) * Lb - Lb
```

The process is repeated for all lines of G-code outputting the translated file in the same directory. The full Python code is shown in Appendix C.2.

4.6 ERROR COMPENSATION

Small misalignments in the print-head hardware can be compensated for. If the movement of either rotational axis causes the unintended movement of the nozzle in any of the linear directions, there is some mechanical imperfection that causes the error. This error can be plotted for a range of rotational axis values, which can then be used to interpolate an error function, which describes how the movement of one axis influences the actual position on another axis. The inverse of this function can then be applied to the IK definition of that axis to cancel out the error, as shown in Equation 4.6.

Table 4.1 Error on X and Y axes by A-axis

Theoretical			*Measured*		*Difference*	
A	*X*	*Y*	*X*	*Y*	*X*	*Y*
-135	79.49	50.49	82.29	48.69	-2.80	1.80
-90	71.00	71.00	72.8	69	-1.80	2.00
-45	79.49	91.51	80.29	90.11	-0.80	1.40
0	100.00	100.00	100	100	0.00	0.00
45	120.51	91.51	121.11	92.92	-0.60	-1.41
90	129.00	71.00	131.2	72.2	-2.20	-1.20
135	120.51	50.49	123.91	51.49	-3.40	-1.00

$$\begin{bmatrix} X \\ Y \\ Z \\ A \\ B \end{bmatrix} = \begin{bmatrix} X' + sin(A') \cdot l_a + cos(A') \cdot sin(B') \cdot l_b \\ Y' - l_a + cos(A') \cdot l_a - sin(A') \cdot sin(B') \cdot l_b \\ Z' + cos(B') \cdot l_b - l_b \\ A' \\ B' \end{bmatrix} - \begin{bmatrix} E_x(A') + E_x(B') \\ E_y(A') + E_y(B') \\ E_z(A') + E_z(B') \\ 0 \\ 0 \end{bmatrix} \tag{4.6}$$

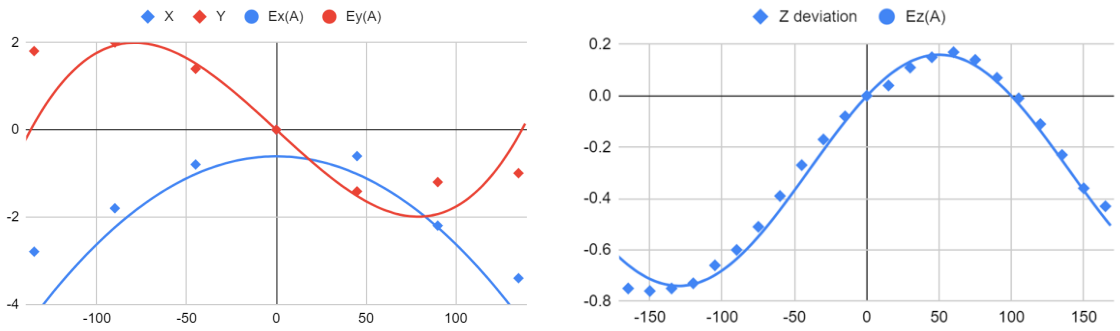
(a) Nozzle tip aligned with calibration cone

(b) After A-axis movement, the deviation in X and Y directions can be found

Figure 4.19: Process for measuring errors in the XY plane

The error introduced by the A-axis rotation on the X and Y axes was measured. This was done by aligning the nozzle's tip with the tip of a reference part, as shown in Figure 4.19a. Then, a certain A-axis angle was requested, which would cause the nozzle to shift slightly from the reference point in the XY plane, as shown in Figure 4.19b. The X and Y axes were manually adjusted until the nozzle lined back up with the reference, and the coordinates were recorded. This process was repeated for a total of 7 A-axis values, in 45°increments. By subtracting the actual X and Y values from the expected ones (as given by the IK formulation), we get the deviation at the given angle, shown in Table 4.1. Plotting the data reveals that the deviations follow a trigonometric pattern, as shown in Figure 4.20a. The error function of the A-axis on the X-axis is given in Equation 4.7, and that of the A-axis on the Y-axis is given in Equation 4.8.

The error introduced by the A-axis on the Z-axis was significantly easier to measure, as a dial gauge



(a) Ex(A) and Ey(A)

(b) Ez(A)

Figure 4.20: A-axis error compensation functions for X, Y and Z found experimentally

could be used to record the vertical displacement caused by the A-axis movement. The dial gauge was positioned normal to the nozzle, as shown in Figure 4.21. A G-code program was then run, which actuates the A-axis in 15°increments on the user's request. The vertical deviations were recorded in the range of -165 to 165, for a total of 23 measurements, shown in Table 4.2. Like before, the results were plotted, as shown in Figure 4.20b. The error function is given in Equation 4.9.

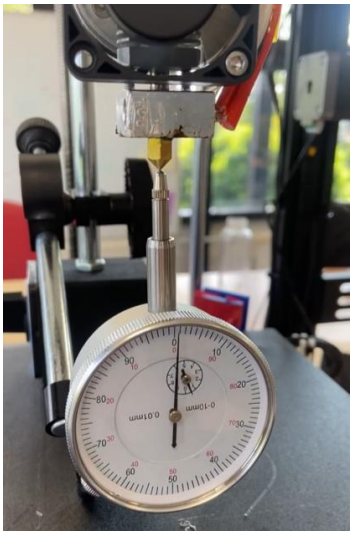


Figure 4.21: Dial gauge used to gather deviation values for Z-axis.

$$E_x(A) = 2 \cdot cos(A) - 2 \tag{4.7}$$

$$E_y(A) = -2 \cdot sin(A) \tag{4.8}$$

$$E_z(A) = -0.45 \cdot sin(A + 40°) + 0.29 \tag{4.9}$$

Table 4.2 Error on Z-axis by A-axis

A	*Z deviation*	*A*	*Z deviation*	*A*	*Z deviation*
0	0	120	-0.11	-75	-0.51
15	0.04	135	-0.23	-90	-0.6
30	0.11	150	-0.36	-105	-0.66
45	0.15	165	-0.43	-120	-0.73
60	0.17	-15	-0.08	-135	-0.75
75	0.14	-30	-0.17	-150	-0.76
90	0.07	-45	-0.27	-165	-0.75
105	-0.01	-60	-0.39		

These same principles of error compensation can be used to further fine-tune the accuracy of a given printer. The values are not transferrable between different builds of the same hardware, as the imperfections are unique to each unit. Additionally, manipulations with the print-head, such as remounting or nozzle changes can cause small deflections requiring a re-calibration to maintain performance. In this case, no error compensation was made for the B-axis due to the lack of an easy solution for measuring deviation.

4.7 SIMULATION

Multi-axis movement carries the risk of collisions between the hot-end and the workpiece, which is impossible in typical FDM as the hot-end only ever recedes from the build-plate, never crossing the current plane of deposition. To prevent such collisions, the full movement of the print-head executing the toolpaths can be simulated.

The simulation is scripted in GH using the *loop* components of the plug-in *Anemone*. These allow loops running within the GH environment, just like the commonly used *for loop*s in programming. The for loop advances the position of the print-head geometry by one point. A collision component continuously checks for collisions between the print-head geometry and the poly-line that represents the past toolpath. The loop has two exit conditions. Firstly, a manual toggle to stop the simulation. Secondly, the output of the collision component is an exit conditions for the loop. Thus, if there is a collision, the simulation would stop. The GH definition for this script is shown in Figure B.1.

Chapter 5

Validation

THIS CHAPTER ASSESSES THE CAPABILITIES of the developed printhead. The performance is evaluated through the printing and examining of sample parts (artifacts). These samples were characterized and compared to samples printed using a typical 3-axis process.

With the overarching research objective to investigate whether the printer can address the shortcomings of conventional FDM without introducing new drawbacks, parts that specifically exemplify the limitations of 3-axis printing were analysed. Thus, first, the rationale is given for the chosen **sample parts**. These parts were then analysed using dense point 3D scanning, to evaluate their **surface quality**, quantifying the improvement attributable to the 5-axis process. Using the dense point cloud, the **dimensional accuracy** of the parts was verified, to see if the multi-axis process had an adverse effect. The **waste reduction** of the multi-axis process was then evaluated based on printing times and part mass.

The validation of the 5-axis retrofit hinges on that it should address the shortcomings of the conventional 3-axis FDM process, without introducing new drawbacks. Its performance must be viewed in *comparison* to the conventional one. Thus the samples parts were also produced using planar slicing, and executed with 3 axes, so that the results could be compared.

5.1 SAMPLE PARTS & PARAMETERS

5.1.1 Geometry

Evaluation of the printer begins with the careful selection of sample parts for benchmarking, commonly known as test artifacts. Section 2.2.1 studied artifacts proposed or used in academic literature, and the lack of any common artifacts for multi-axis FDM. Artifacts are often tailored to the boundaries of the process that they're used to evaluate. Therefore, those artifacts used for typical AM processes, including conventional FDM, are not best suited for evaluating the capabilities of a 5-axis FDM printer. The two geometric benefits of the multi-axis approach are the ability to print unsupported overhanging geometry; and reduced aliasing in oblique surfaces, particularly at shallow angles to the build-plate. Ideally, the parts should be usable to ascertain the printers performance in these two aspects with granularity, so they should have a wide range of overhanging geometry and oblique surface angles. Thus the following main criteria were selected for the sample parts:

- Wide range of overhanging geometry angles, up to 90°(horizontal); and

- full range of oblique surface angles.

The designed sample parts to fit the criteria were designed as a circular profile, swept using 90°arc-shaped rails. Two of these parts - AB-tube and B-tube - were used, to isolate the errors of the two individual rotational axes, giving more insight into the inaccuracy introduced by each. The AB-tube is curved around both Y and Z axes, requiring both A- and B-axis movement to print. Whereas the B-tube is only curved around the Y-axis, therefore not requiring any A-axis movement. The CAD data of the parts, shown in Figure 5.1, was parametrically generated using the GH definition in Figure B.2.

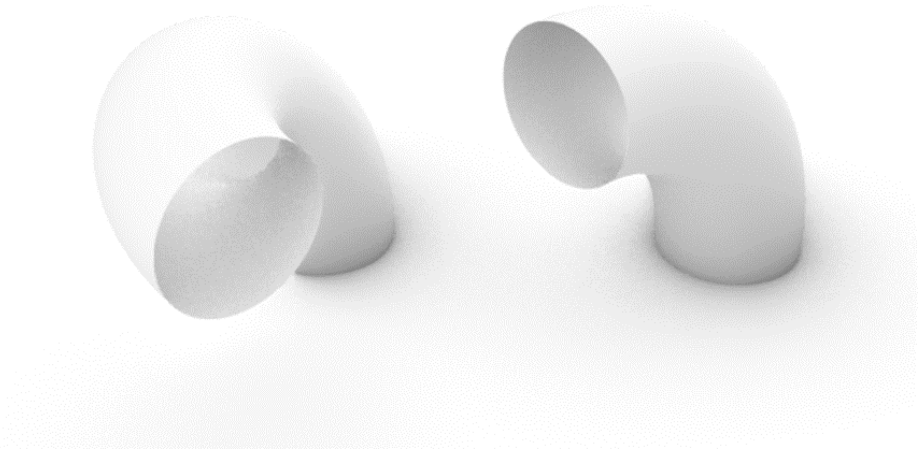


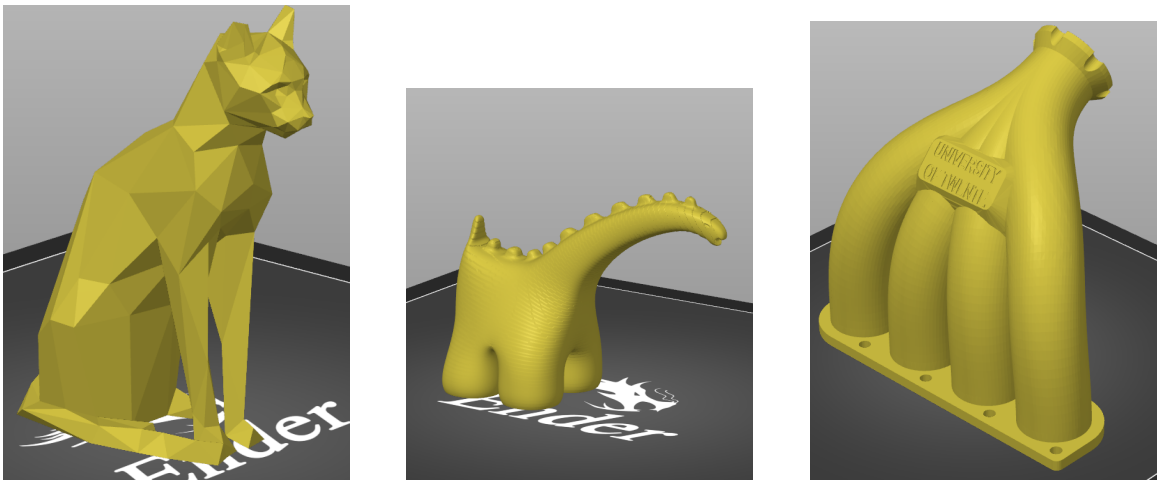
Figure 5.1: Sample parts AB-tube and B-tube

While these quantitative test artifacts serve as critical benchmarks for evaluating the printer's capabilities, it is equally important to validate their practical relevance in real-life applications.[97] To achieve this, additional sample parts with a broader variety of shapes. E.g. the Sitting cat and Dinosaur models, shown in Figures 5.2a and 5.2b respectively, are representative of aesthetic models or trinkets often produced using FDM, exhibiting complex surfaces and fine features. Meanwhile, the Exhaust manifold, shown in Figure 5.2c, is representative of functional parts, including features such as internal cavities, bores, and mating surfaces.

By combining the quantitative assessment facilitated by the AB-tube and B-tube test artifacts with the qualitative evaluation enabled by the sitting cat and exhaust manifold, we gain a comprehensive understanding of the 5-axis FDM printer's capabilities. The synergy between these two approaches reinforces the significance of the printer's unique geometric advantages while demonstrating its applicability to real-world scenarios.

5.1.2 Slicing parameters

For an objective comparison between the planar and non-planar printed parts, ideally the sole difference would be the enabling of the additional axes. However, such a direct comparison is not possible to execute on the sample parts, as planar slicing can not produce perimeter-only parts with such extreme overhang angles. Overhangs require a combination of perimeters and bridging



(a) Sitting cat[98] (b) Dinosaur (c) Manifold

Figure 5.2: Sample parts with complex geometry

infill toolpaths, as shown in Figure 5.3. Instead, the approach will focus on getting the best results out of both processes, while maintaining the critical slicing parameters the same. While it is well established that layer height and surface angle are major driving factors of surface roughness. The surface angles are enforced by the CAD data, which is the same for both processes. The layer height on the 5-axis process is variable, so the average of that is used for the 3-axis process.

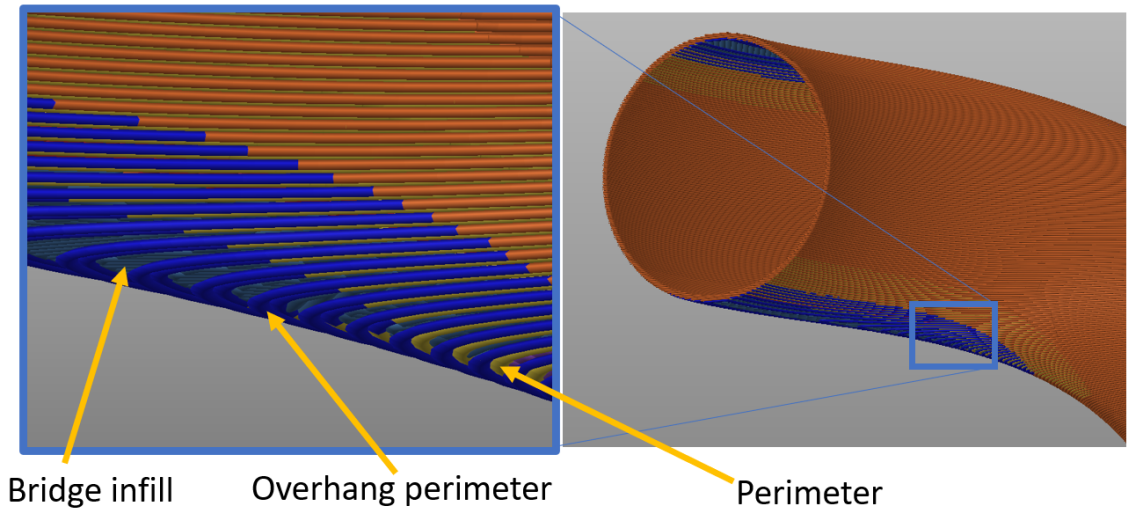


Figure 5.3: Composition of planar overhang toolpaths, overhanging toolpaths shown in blue

5.1.2.1 Planar parameters

Initial attempts to print planar utilized the `0.2mm SPEED @MK3` preset profile in PrusaSlicer, however, that resulted in support structures being overly bonded to the part. Structures supporting the outer wall of the tube could be removed, but those nested inside the tube could not be removed without damaging the part, as shown in Figure A.4. Two modifications were made to the options

for support material to remedy this - `Style` was set to snug[1], and `Top contact Z distance`[2] was increased from the default 0.2 mm to 0.3 mm. The resulting internal support structures could now be removed without breaking the part. This illustrates the difficulty in directly comparing the 3- and 5-axis processes, as both can be tuned to achieve better results depending on the specifics of the printable geometry.

5.1.2.2 Non-planar parameters

For the non-planar slicing, the Iso-curves slicing algorithms was used, explained in Section 4.2. This path-planning algorithm lends itself well to the slicing of surfaces, such as the swept profile, producing them using a single-walled spiral, akin to vase mode on conventional slicers. This simple heuristic means that there are few critical slicing parameters - nominal layer height, track width, print speed, and resolution. Some parameters relate purely to the filament used, such as filament diameter and print temperature. Other parameters include modifiers to the flow, to fine-tune the result. The detailed overview of parameters for this algorithm are explained in Section 4.2. For the purposes of printing the sample tubes, a nominal layer height was found that results in a maximum layer height $\leq 0.3mm$. That is the largest recommended layer height for 0.4 mm nozzles, which are the standard size in consumer FDM printers.

5.1.3 Printers

For the most direct comparison both planar and non-planar samples were printed on the same modified 5-axis printer, just using planar and non-planar slicing respectively. However, the planar prints were not coming out nicely, due to the lack of part-cooling. Printing non-planar without part cooling lead to extremely poor support interface, as the interface layers were blending together with the part, making it impossible to remove them. Therefore, the non-planar samples were instead printed on a Prusa i3 MK3, which is a similar but slightly more premium desktop FDM printer. While this makes it impossible to isolate the differences between the printer's linear axes, both printers represent the same class of printers.

5.2 DIMENSIONAL ACCURACY

A key concern is that the addition of axes to the printer would reduce the accuracy of the machine, and thus the accuracy of the parts that it produces. Theoretically, adding the two rotational axes in series to the printhead increases the kinematic error at the tip of the nozzle. As one of the objectives of the work was to investigate whether the design would introduce drawbacks to the FDM process, such as a reduction in the dimensional accuracy of printed parts. The A-axis is expected to incur a greater error than the B-axis, since the axis itself has a lower accuracy due to the lack of mechanical reduction compared to the B-axis, as explained in Section 3.1.3. Part scale is not expected to have any effect on its dimensional accuracy, because the error of all axes (linear and rotational) is constant throughout the build volume, as explained in Section 3.1.1.

[1] Less stable, but reduce scarring.
[2] Vertical distance between object and the interface of the support material.

5.2.1 Methodology

The validation of dimensional accuracy was performed using dense-point 3D scanning. In this process, a 3D scanner is used to produce an accurate mesh of the sample. Then, the captured mesh can be compared to the theoretical CAD data, that was used to print the part. Metrology software is used to fit the scanned mesh data to the CAD data, to find the deviation between the two.

Figure 5.4: Setup for dense point 3D scanning with EinScan Pro 2x industrial kit

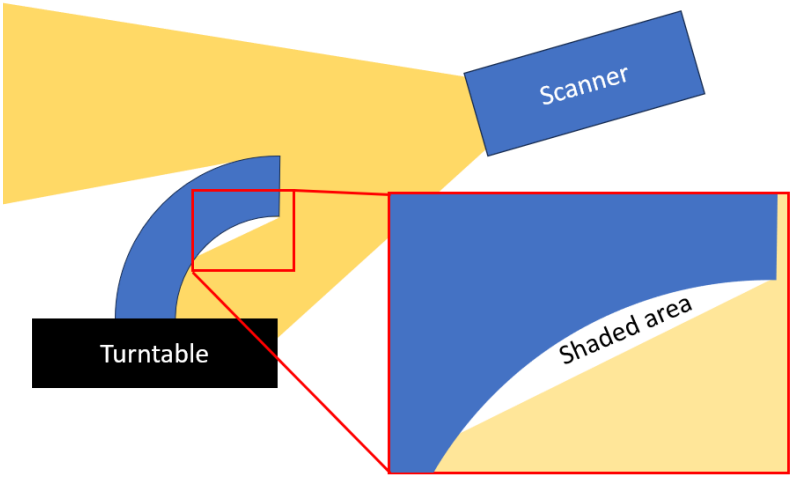


Figure 5.5: Schematic sideview of poor line-of-sight for overhanging geometry

The EinScan Pro 2x handheld hybrid laser scanner by Shining 3D was used together with a tripod and markered turntable (their industrial kit), in the setup shown in Figure 5.4. The use of a tripod and turntable enables "fixed scans", which yield the highest accuracy for the scanner - 40 µm. This is thanks to the scanner being able to perform a long exposure scan at each of the angles with plenty of markers visible, which is not possible if used hand-held due to the lack of stability. Handheld scans reduce the accuracy to 200 µm. A limitation of the fixed scanning is that the turntable's markers must be clearly visible to the scanner. Thus the scanner should be at least slightly angled downwards, toward the turntable. This limits the ability to scan overhanging geometry, as the

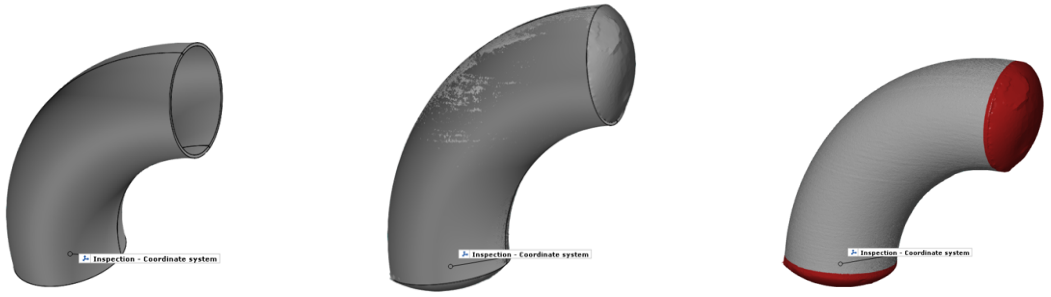
bottom of it would not be visible to the scanner, as shown in Figure 5.5. Performing additional scans with the part in a different position would introduce some error, as the separate scans would then have to be aligned. A 45°pedestal was designed for the tubes to circumvent the issue. The 45°angle makes all the outer surfaces of the sample tubes fall within the scanner's line-of-sight, as shown in Figure 5.6. Internal surfaces of the tubes are largely obscured, making it impossible to get a full picture of the accuracy of the internal surface.

(a) Full line of sight for inside bend

(b) Full line of sight for outside bend

Figure 5.6: Improving the scanner line-of-sight using 45°pedestal

GOM inspect is a software for analyzing 3D data, used for part inspection and quality control. The CAD data, shown in Figure 5.7a, can be imported alongside scanned data for a part for comparison. Both geometries are automatically aligned, as shown in Figure 5.7b, and the deviation between them can be found at any point. Parts of the mesh at the ends is excluded from the analysis, as there are errors in the mesh, as shown in Figure 5.7c. A *surface comparison* can then be produced, which maps the dimensional deviation as a heat-map, and produces values for the maximum, minimum, and mean distance. These values serve as quantitative measurement of the average deviation, and the range of deviation. The heat-map can be further examined to make judgements regarding deviations in certain specific areas of the samples. *Inspection sections* can be produced, giving additional insight into specific cross-sections of the part.



(a) CAD model of theoretically perfect part

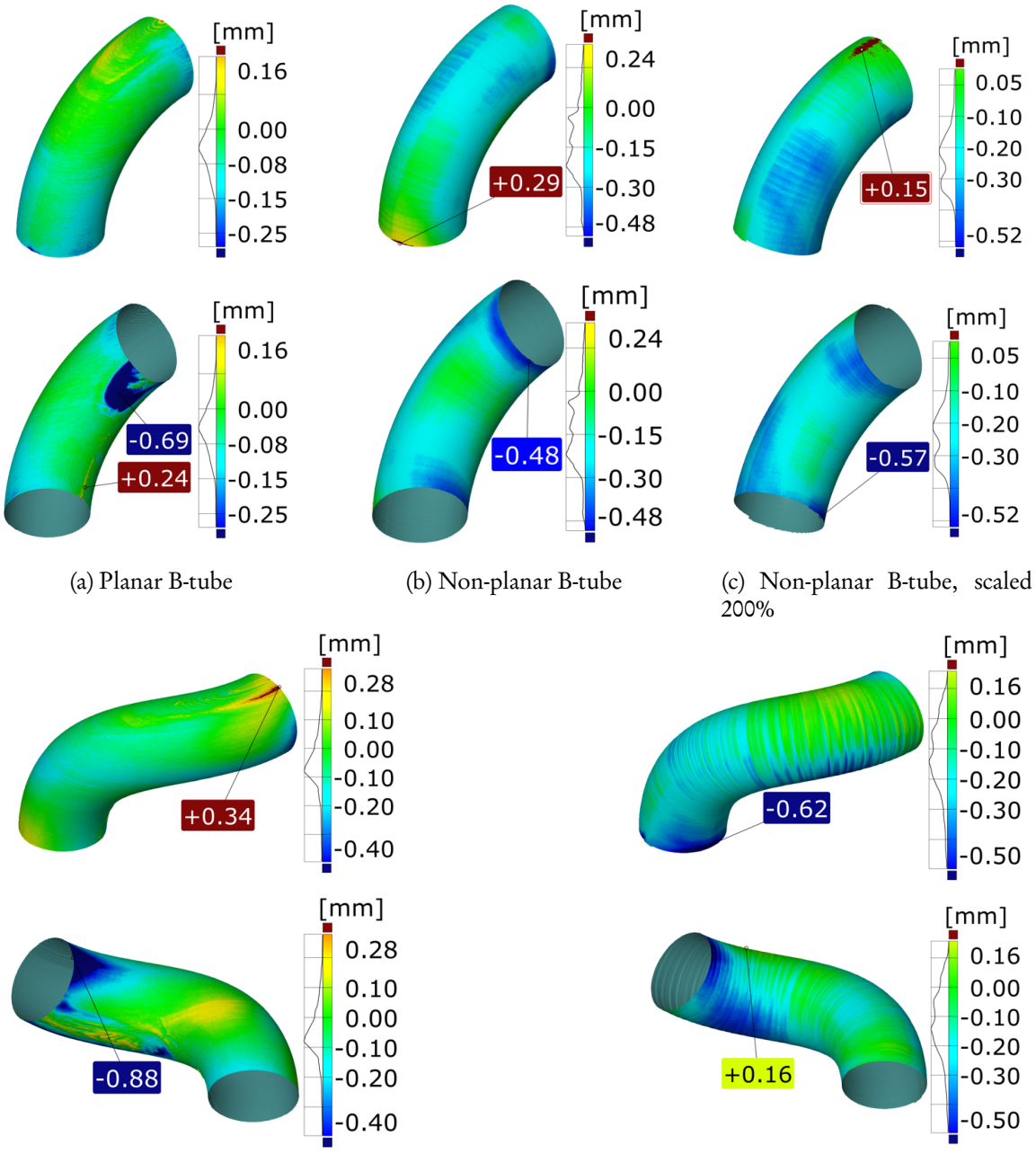
(b) Scanned mesh fit to CAD data

(c) Trimming of incorrectly meshed hollow ends of sample, marked in red

Figure 5.7: Processing steps for preparing CAD data and scanned mesh for surface comparison in GOM

5.2.2 Results

5.2.2.1 Surface comparison



(a) Planar B-tube

(b) Non-planar B-tube

(c) Non-planar B-tube, scaled 200%

(d) Planar AB-tube

(e) Non-planar AB-tube

Figure 5.8: Surface comparisons between scanned meshes and CAD models

Figure 5.8 shows the surface comparison for all samples. The minimum and maximum deviation points from nominal are shown, alongside a histogram showing the distribution of the deviations. The results for the range of deviation and mean deviation are compiled in Tables 5.1 and 5.2, and

Table 5.1 Range of deviation between scanned meshes and CAD models

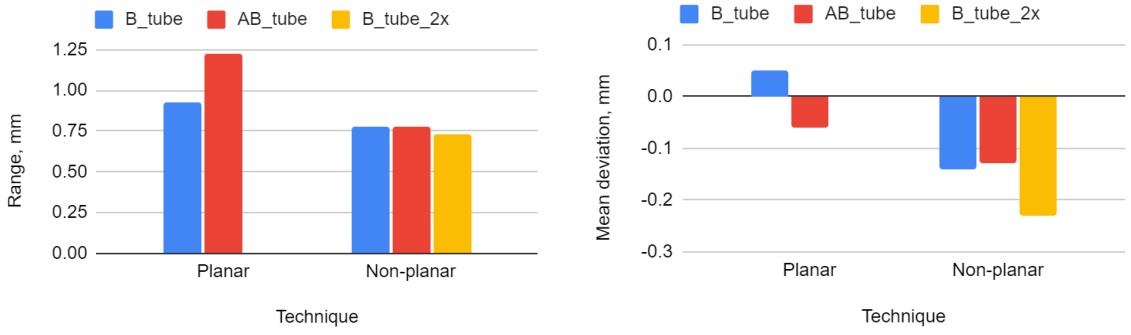
	B_tube	*AB_tube*	*B_tube_2x*
Planar	0.93	1.22	
Non-planar	0.78	0.78	0.73

Table 5.2 Mean deviations between scanned meshes and CAD models

	B_tube	*AB_tube*	*B_tube_2x*
Planar	0.05	-0.06	
Non-planar	-0.14	-0.13	-0.23

shown in Figure 5.9. The data indicates that the range of deviations for the non-planar samples is lower than that of the planar samples and more stable. However, the mean deviation of the non-planar samples is higher. In general terms, the two processes are comparable in terms of dimensional accuracy and excel in their own ways. Furthermore, the double-scaled sample shows no significant reduction in accuracy compared to the respective nominal-sized sample.

Looking at the planar B-tube sample in Figure 5.8a, we can see that the extremes of the deviation can be clearly related to artifacts resulting from the planar path-planning process. The maximum deviation is on the seam line on the print, which is where each layer's perimeter starts. Due to the travel between printing the perimeter and support structure, there is a small build-up of excess material from imperfection in the retraction of filament. The minimum deviation is on the supported section of the overhang. To ensure the support structures can be easily removed, a small airgap is present between the support interface material and the perimeters. For this reason, keeping tolerances tight for even supported overhanging geometry is difficult, as the exact extent of material drooping can be difficult to predict. As prior mentioned, the Z airgap was increased from the default, which contributes to this inaccuracy. However, it was necessary, as the manual post-processing of tighter supports caused damage to the samples. It should also be noted that the planar AB-tube sample, shown in Figure 5.8d, does not have its extreme deviations in the same locations, even though the supported geometry appears to be affected.



(a) Range of deviation

(b) Mean deviation

Figure 5.9: Graphed data for deviations between scanned meshes and CAD models

5.2.2.2 Section inspection

Section inspections were produced for the B-tube samples, as shown in Figure 5.10, as they can be cut in half by a plane. Examining the profiles of the samples in Figure 5.11 shows that the deviations get larger in amplitude towards the shallower incline of the oblique surfaces of the planar sample. Due to the magnitude of surface roughness, it is possible to see trends in roughness even on the relatively low resolution of the 3D scans. More in-depth characterization follows in the next section.

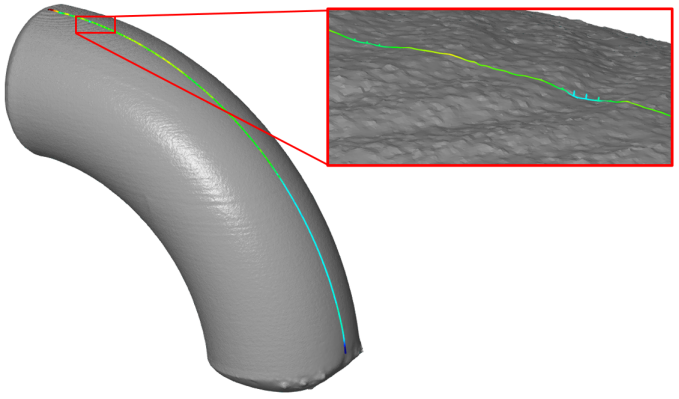
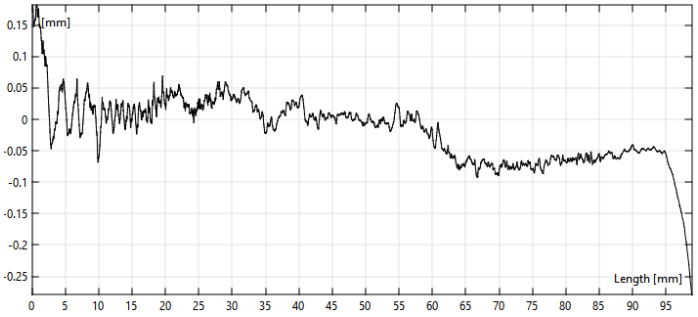
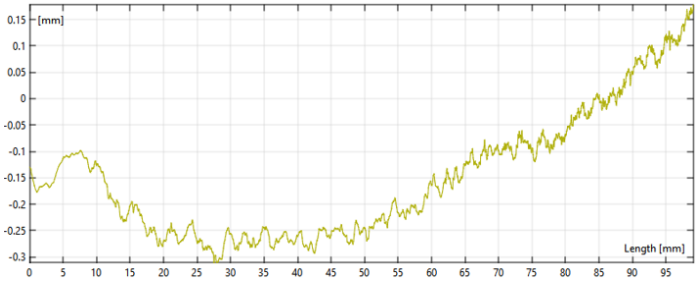


Figure 5.10: Section inspection of planar B-tube sample in GOM inspect



(a) Planar sample showing increasing amplitude of inaccuracy



(b) Non-planar sample showing consistent amplitude of inaccuracy

Figure 5.11: Section profiles for B-tubes, extracted using section inspection in GOM inspect

5.3 SURFACE QUALITY

One of the key claims of the benefits of multi-axis FDM is improved surface quality, due to the reduction of aliasing. At any angle that is not 90°, planar toolpaths should yield a rougher surface, and this effect is particularly prevalent at shallow angles. The previous section analysis of the 3D-scanned surface allowed us to see this trend, however, the resolution is not sufficient to measure the surface roughness.

The surface roughness of the non-planar samples is expected to remain constant along a given vertical iso-curve, whereas it should be increasing on the 3-axis sample. The surface is expected to be rougher on the outer bend of the 5-axis sample than the inner, due to the variable layer height.

Surface roughness is typically characterized as the arithmetic mean deviation from the neutral profile, which is essentially no different from the accuracy characterization done in the previous section. The difference stems from what is considered to be surface roughness, versus waviness or profile.[3] Typically such large surface imperfections would not be classified as roughness. However, in the context of this study, we consider this to be the primary attribute of *surface quality*. The roughness is typically expressed as Ra for 2-dimensional data and Sa for 3-dimensional.

5.3.1 Methodology

The validaton of the surface quality was performed using confocal laser scanning microscopy, also known as simply confocal microscopy. In this process a laser light source is used to scan the sample at a particular focus distance. The process is repeated at many focus distances with small increments. The result is a high resolution 3D image of the surface - a topographical map. Various surface parameters can be extracted from the data. In this case, the arithmetic mean height deviation was used to characterize the surface roughness of samples.

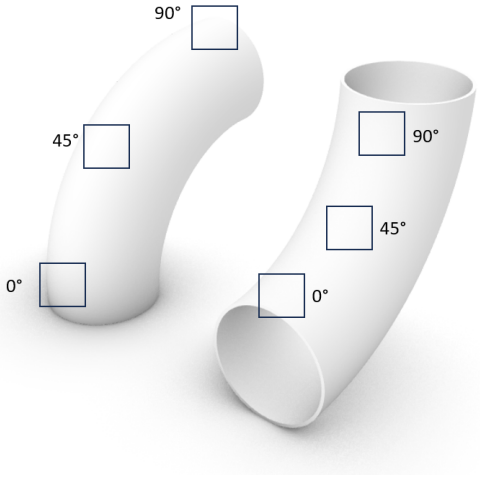
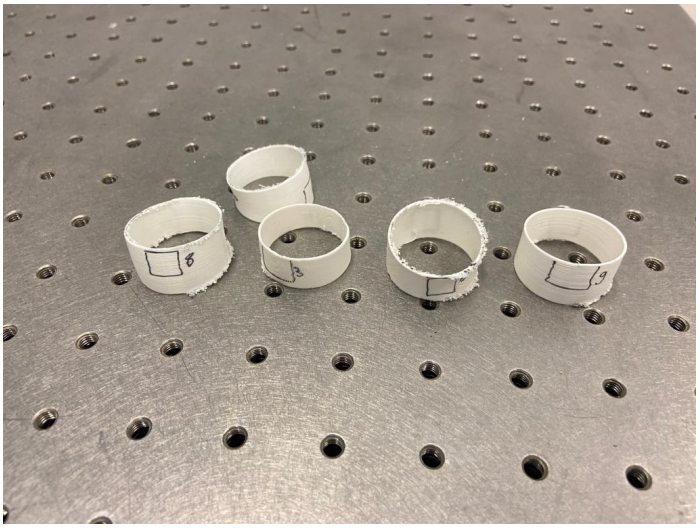
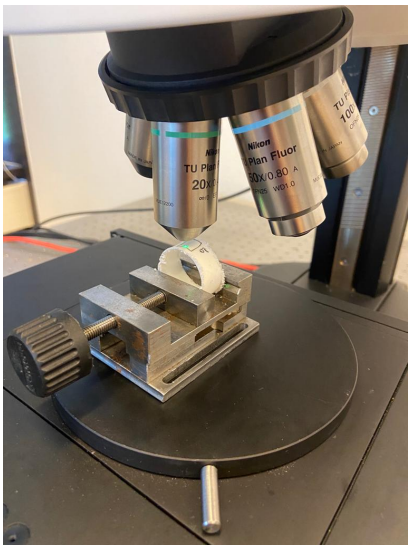


Figure 5.12: Areas for sampling surface roughness

[3] There is no strict definition. In Earth Sciences surface roughness is measured in meters.



(a) Cut samples



(b) Fixtured sample

Figure 5.13: Sample preparation for confocal microscopy

Measurements were performed using the Sensofar S Neox non-contact 3D opical profiler. For the planar and non-planar samples, small areas for measurement were marked at about 0°, 45°, and 90°overhang, on both the inside and outside bend, as shown in Figure 5.12, for a total of 12 measurement areas across the two B-tube samples. Fixturing the samples for microscopy required cutting them into smaller pieces, as shown in Figure 5.13a, which allowed them to be fixtured as shown in Figure 5.13b. Considering the large roughness of FDM substrates, a wide Z-scan range of 500 µm was used. At a step or accuracy of 1 µm, 500 planes of each sample were scanned to achieve each 3D image. Some noise is present in the data, possibly due to subsurface scattering in the polymer substrate, causing difficulties getting perfect focus. The SensoVIEW software was then used to analyze the data.

5.3.2 Results

The extracted arithmetic mean deviations Sa are shown in Table 5.3, and graphed in Figure 5.14. From this we can see that the surface roughness of the non-planar samples remains within 19 to 30 µm - a range of just 11 µm. Whereas the planar samples have roughness ranging from 22 to 114 µm. Furthermore, the inside bend of the non-planar samples has a lower roughness, as predicted. Roughness on the planar samples drastically increases for the top area, which is the shallow oblique surface. The bottom inside sample is an outlier, with a higher roughness than expected. This is due to the presence of a seam, exacerbated by the excess residue from travels from the support-building.

Figure 5.15 shows the sampled surfaces on the planar B-tube sample, Figure 5.16 shows the non-planar. Assessing the topographical maps of the planar samples visually, the top surfaces are highly irregular compared to the side and oblique surfaces. The non-planar samples are more regular throughout the length of the arcing axis. However, the non-planar samples appear to exhibit a pattern, where every other layer is somewhat recessed. This is likely due to the low local layer height compared to the track width.

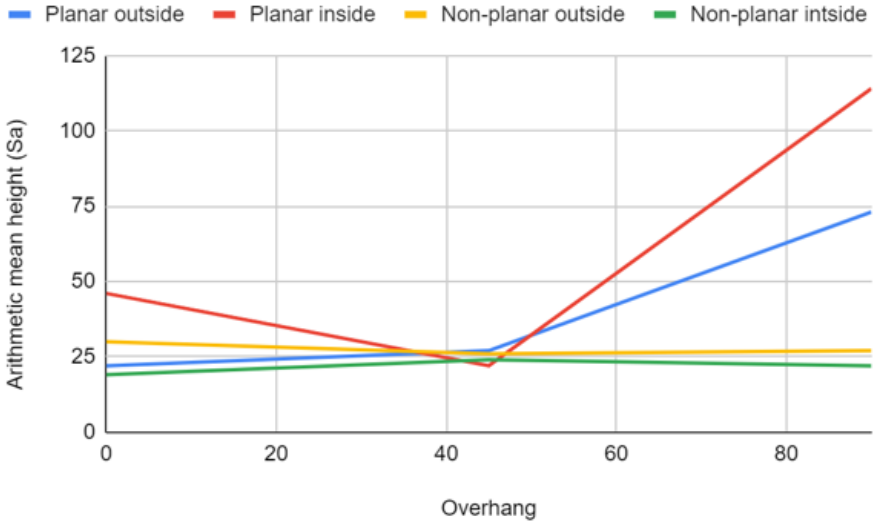


Figure 5.14: Surface roughness of B-tube samples. Data points connected to illustrate trend.

Table 5.3 Surface roughness of B-tube samples

	Planar		*Non-planar*	
Overhang	*Outside*	*Inside*	*Outside*	*Inside*
0	22	46	30	19
45	27	22	26	24
90	73	114	27	22

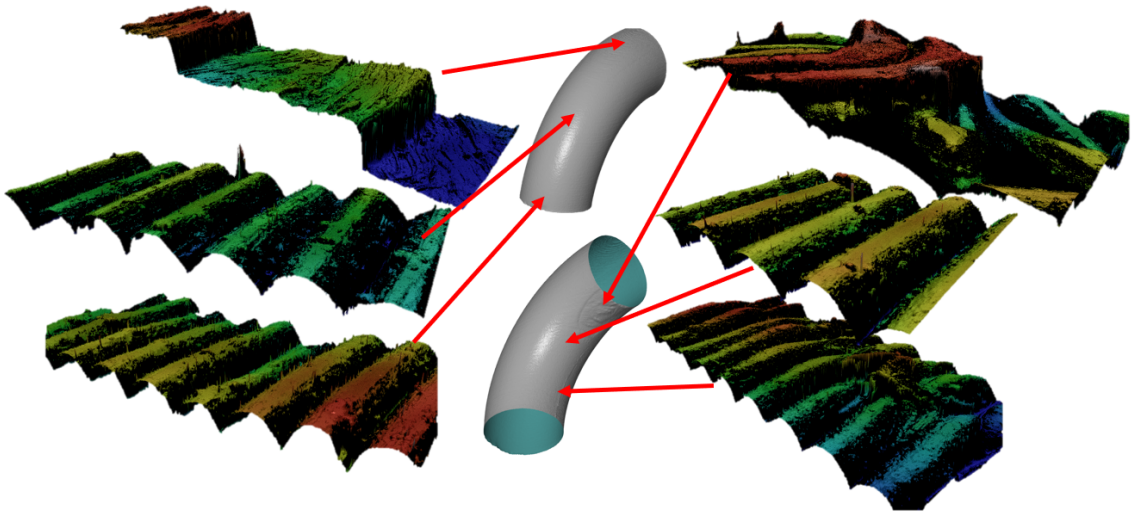


Figure 5.15: Topographical surface maps of sampled areas on planar B-tube

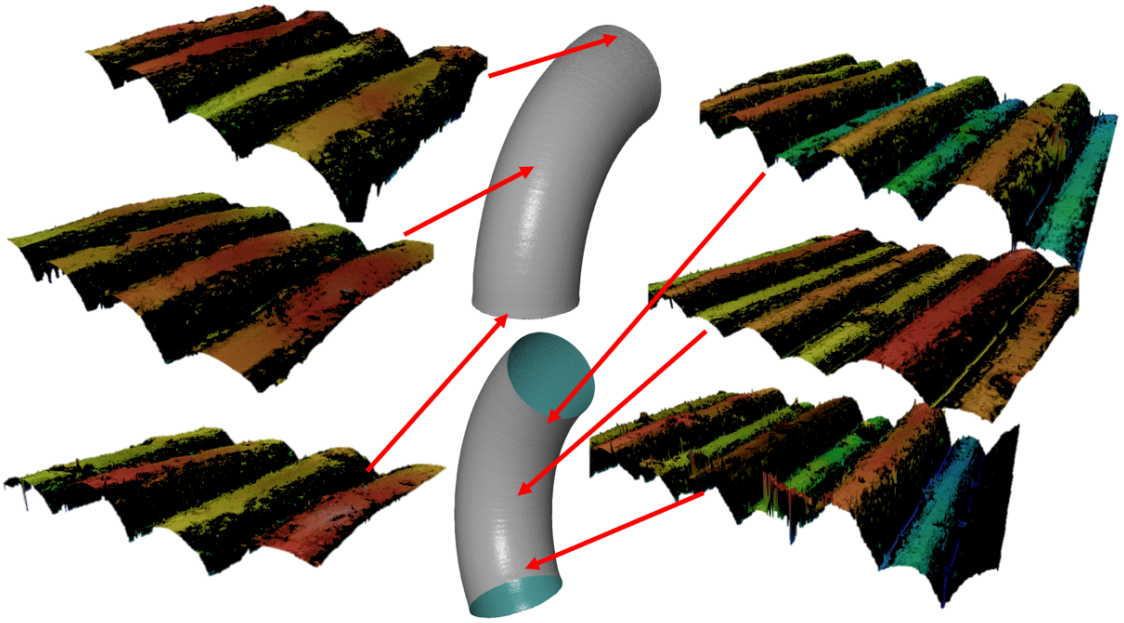


Figure 5.16: Topographical surface maps of sampled areas on non-planar B-tube

5.4 WASTE REDUCTION

Multi-axis enables printing overhanging geometry without needing support structure - support-free printing. The waste-reduction associated with this ability is multi-fold. For one, there is a reduction of waste material, which would have been used to build the sacrificial support structures. Furthermore, including these support structures significantly increases the necessary printing time of parts. Furthermore, the removal of the support structures is a manual post-processing step, the difficulty of which varies widely depending on the specific geometry. Additionally, support structures may cause undesired artifacts on the parts, such as a more pronounced seam or stringing, which also require additional post-processing to clean up. The use of soluble support structure material, such as water-soluble PVA, is possible when quality is critical. However, it is an expensive material which also requires special handling,[4], requires a more sophisticated machine capable of multi-material printing, and requires the post-processing step of bathing and drying. Finally, the removal of support-structures has a risk of damaging the part, in which case the entire printing process must be repeated, which occurred on several occasions attempting to print the planar samples.

The material and printing time reduction is trivial to quantify for the test samples, as both can be calculated at the slicing stage of print-job preparation. On the other hand, the labor costs associated with post-processing, and full part reworks resulting from poor post-processing could be measured, but is not particularly informative, as both are highly dependent on specific part geometries and delicacy of the part. Instead qualitative judgements can be drawn.

Table 5.4 shows the material usage and print-time of the sample parts used in characterization of

[4] As a hygroscopic material it must be kept dry from the atmospheric humidity.

Table 5.4 Material and print-time waste comparison

	Planar		*Non-planar*		*Improvement*	
	Print time (hh:mm)	*Material use (g)*	*Print time (hh:mm)*	*Material use (g)*	*Print time (% reduction)*	*Material use (% reduction)*
AB_tube	3:25	31.2	0:30	9.2	85.37%	70.51%
B_tube	2:12	19.9	0:21	6.8	84.09%	65.83%
B_tube_2x	12:41	140.2	1:14	27	90.28%	80.74%
Manifold	24:04	135.1	11:28	61.1	52.35%	54.77%
Cat	3:41	32.6	3:01	25.5	18.10%	21.78%
Dinosaur	4:17	24	4:00	17.3	6.61%	27.92%

dimensional accuracy and surface roughness, alongside other parts to represent a wide variety of geometries and applications. In the extreme cases, multi-axis printing can reduce the print time by as much as 90%, and the material usage by as much as 80%. The reason print-time doesn't scale linearly with material usage is that the printing of support materials also incurs an additional time penalty for travel between the support structure and part. Considering a part that has a very low amount of overhanging geometry - the sitting cat - a respectable print-time reduction of 18% and material reduction of 21% was reached.

5.5 SUMMARY

The purpose of the validation was to show that the developed print-head can produce the improvements in surface quality and waste reduction, which are touted as two of the core benefits of multi-axis FDM; but also to verify that the dimensional accuracy has not been compromised, which is a key concern with multi-axis FDM.

Studying the surface quality of samples revealed that the non-planar samples had superior quality, maintaining a low roughness throughout the full range of surface angles. Unlike the planar samples which exponentially worsen towards shallow angles. In terms of waste reduction, the geometry of the part has a massive impact on the result. The ratio of part to support structure, the area of overhang, as well as the overall size, all impact the result. While the most extreme scenarios showed a time and material reduction of over 90% and 80%, respectively, one of the more conservative scenarios suggests a 20% reduction for both. Both of the studies offer strong evidence that these two benefits of multi-axis FDM can be attained using the developed accessible print-head.

Studying the dimensional accuracy revealed that the range of inaccuracy for the non-planar samples are not just on par with that of the planar samples, but even lower and more consistent, which can be attributed to the lack of certain printing defects inherant to planar slicing. However, the mean deviations are higher for the non-planar samples, in the order of about one layer height. This indicates that no significant penalty has been incurred in the dimensional accuracy, however, some negative effects are present.

Chapter 6

Conclusions

AFTER THE THOROUGH EXPLORATION of the hardware, software, and results of the created multi-axis solution, this chapter reflects on the key findings and the answers to the **research questions** initially posed. Furthermore, potential directions for **further work** are laid out. Finally, we look back at how the work fulfills the overarching **research objective**, concluding this work.

6.1 RESEARCH QUESTIONS

6.1.1 Surface quality

A key shortcoming of conventional Fused Deposition Modeling (FDM) is the notoriously poor surface quality, characteristic of the discrete layer lines, particularly for shallow angle oblique surfaces. As one of three core benefits of multi-axis FDM is the improvement in this surface quality, the first research question posed was *"how does multi-axis FDM compare to conventional FDM in terms of surface roughness?"* The journey to answering this question led all the way through the development of the printer hardware, controls, sample selection, path-planning algorithms, calibration techniques, and ultimately, characterization.

A sample part with various degrees of surface angle was printed using a popular conventional FDM printer, and one using the developed retrofit print-head, to validate the hypotheses: The surface roughness of planar prints increases toward a shallow angle[1], but remain the same on non-planar prints; and the surface roughness on non-planar prints is higher on the outer bends. Considering the significant roughness of the FDM process, the answer could be ascertained even through inspection by the naked eye or physical touch. Furthermore, 3D scan data acquired for the accuracy characterization was also corroborating the findings. Nonetheless, confocal microscopy was performed on 12 sample areas across both samples to acquire quantitative data of specific roughness values. The analysis showed that non-planar samples had a consistent surface roughness, staying within a tolerance of 3 um and 5 um for the outside and inside bends respectively. Additionally, the inside bend showed a slightly reduced roughness compared to the outer. Both findings for the non-planar sample agree with the theory. On the other hand, the planar samples exhibit a strong increase in surface roughness towards the shallow surface angle. At its worst, the planar print has a surface roughness in excess of 5 times that of the non-planar. The results are shown in Section 5.3.2.

[1]Shallow angle relative to the build-plate.

6.1.2 Waste reduction

Another key shortcoming of conventional FDM is the inability to print unsupported overhanging geometry, requiring sacrificial support structures to be built. This incurs a waste of physical material, machine time, and labor of manual post-processing. Multi-axis' ability to print in any direction enables support-free printing eliminating these wastes. Thus the next research question followed: *"How much waste - material and time - can be reduced by using multi-axis FDM to print parts with overhanging geometry?"*

The material and machine time waste was quantified using theoretical calculations produced as part of the path-planning process.[2] The analyzed print jobs were also executed, to verify that the result is attainable. The analysis was performed both on the same sample parts that were used for accuracy and surface roughness characterization, as well as a number of more complex shape parts representing both aesthetic and functional parts. Results indicate, in the most extreme case, a print time and material use reduction in excess of 90% and 80% respectively. Whereas the scenario with the least support structures still yielded an improvement of more than 18% and 21% respectively. These reduction of waste are most prevalent for those parts that have the highest ratio of support structures to part geometry, such as the thin-walled tubular parts. When examining solid body geometry, which is more representative of typical FDM use-cases, the improvement was less extreme.

6.1.3 Dimensional accuracy

The improvements in surface quality and waste reduction having been proven, the next question was whether any gain in those areas has inadvertently impacted other performance metrics. A key concern in the design of a multi-axis printer is the increased error due to the additional elements in the kinematic chain. The additional error could cause dimensional inaccuracy rendering other benefits moot. Thus follows the question: *"How does a multi-axis FDM process compare to a conventional one in terms of dimensional accuracy?"*

The dimensional accuracy was characterized using dense point 3D scanning of entire sample parts. The produced mesh geometry was then compared to the theoretical CAD data used to print the parts. Using specialized metrology and inspection software, the deviation between points on the scanned mesh and the nominal positions was found. This particular study has some overlap with the surface roughness characterization, as surface roughness is also a form of inaccuracy. As was mentioned earlier, the roughness of FDM surfaces is significant enough to be perceivable on the scale used for 3D scanning. The analysis found that the multi-axis prints have a greater mean deviation, on the order of about one layer height. However, the range of deviations was found to be both lower and more stable across different multi-axis prints. Additionally, the test was used to verify that the scale of the part doesn't decrease accuracy, as was asserted in Section 3.1.3 regarding the differences between head-head (HH) and table-table (TT) configurations for the rotational axes. While the range of inaccuracy was on par with the smaller samples, unfortunately, the mean inaccuracy was increased, which warrants additional investigation.

[2] Excluding the build-plate and nozzle warm-up ahead of the print job, which is equivalent regardless of process DoF.

6.1.4 Design parameters

A fundamental aspect of the work done is the investigation of design parameters to successfully produce the accessible multi-axis printer. Various designs have been shown in academic literature, and some in the commercial and enthusiast space. Many show no rationale for design decisions made, which explains the failure of many to produce parts of sufficient quality, posing the question: *"What design parameters of a multi-axis 3D printer are critical for achieving (the) benefits, and what can be done to improve results further?"*

Chapter 3 is a deep dive into the mechanical design of the developed solution, which is a retrofit print-head compatible with a wide range of commercial off-the-shelf (COTS) printers, including the popular and affordable Ender 5 by Creality.[3] The choice of developing a print-head (HH configuration) as opposed to a rotary table (TT configuration) is justified based on an analytical study of how each would affect the build volume, accuracy, and accessibility. The HH configuration for the rotational axes was found to have a larger build volume given the same hardware dimensions, which is crucial when other multi-axis designs have build areas significantly smaller than typical for consumer printers. The accuracy was also analytically found to be consistent across the build volume, which would reduce the deterioration of print quality for larger parts. The prototyping and testing of the print head revealed further lessons on how specific part geometry influences the printer's reach. The limitations of using a retrofit print-head on a host printer were also considered, as there are limitations in clearance that are inherent to the kinematic structures typically used for FDM. Portal gantry style kinematics, commonly used for CNC mills was recommended to get the most out of the 5-axis functionality.

6.1.5 Path-planning algorithms

The validation of the printer design requires the printing of sample parts that can be analyzed to evaluate the resulting performance of the printer. However, generalized path-planning tools for multi-axis printing are not widely available. Thus, in order to produce G-code for my printer I would have to identify the applicable algorithms for slicing the sample parts, prompting the question: *"What path-planning algorithms can be used to effectively print the chosen benchmark artifacts, and how can they be implemented?"*

For the tubular parts used for quantitative characterization, the iso-curves slicing method, as described by Li et al,[79] was found to be suitable. While rudimentary, it offers a simple heuristic to automatically identify non-planar layers on surfaces. The method was implemented using the visual scripting language Grasshopper (GH) within the Rhinoceros 3D CAD software. The general principles of G-code generation was implemented as described by Cuevas,[33] and augmented with capability for rotational axes by including a tool/nozzle vector at each point. The end result is a full GH script that can automatically generate 5-axis toolpaths for surface parts using simple user-input parameters, such as nominal layer height[4], track width, and print speed.

For the complex parts the G-code anamorphosis technique of bending was used, adapted from the work of Hermann.[81] Adding output for additional axes to his Python script allowed the augmentation of his method to support a far wider range of bending angles.

[3] Price varies, but is generally about €300.
[4] Actual layer height is variable and localized.

6.2 FURTHER WORK

The exploration of multi-axis 3D printing is boundless, and this work only scratches the surface. There are specific next actions that could be taken to improve upon the work that was done as part of this thesis.

6.2.1 Improved print-head

Despite going through 4 iterations of print-head design, and ending up with a capable concept, there are certainly improvements to be made, both in terms of additional features as well as enhanced performance. One crucial limitation of the current design is the inability to perform full A-axis rotations, as the electrical wires and Bowden tube would get tangled. This hardware limitation can be worked around by ensuring that the print jobs don't require full A-axis rotations, or having the print-head unwind itself once it reaches the rotational limit. It should be noted that this is not an inherent limitation of the HH configuration, as on the TT configuration the print-bed still needs power and signal cables.[5] Passing the electrical connections through continuous rotation is as simple as implementing a slipring, and a similar slipping connection could be designed for the Bowden tube. This change would also require additional changes in the firmware configuration, as the continuous rotation of the axis would cause the A coordinate to cascade. The option of enabling relative coordinates for the rotational axes in Marlin could be used.

As was noted from the dimensional accuracy characterization, the mean deviation produced using the print-head is slightly lower than that of the conventional printer used for comparison. Thus some improvements could be made in the accuracy of the design. Namely, the A-axis could use reduction through a belt-drive, same as the B-axis. This would also enable the use of the previously mentioned slipring and slipping Bowden connector, as they could be passed through the bearing.

6.2.2 Bending slicer integration

The G-code anamorphosis method of bending is a simple but effective way of getting the benefits of multi-axis for a wide range of geometries. To make it more accessible, a common slicer could be forked to integrate the algorithm into the interface already familiar to millions of users. Common slicers are already starting to include basic CAD functionality, so it is not a stretch to think that users would be interested in G-code bending tools straight in their slicer.

6.2.3 Collision prevention

Section 4.7 described the approach for collision avoidance, which the user can effectively apply to make sure the generated toolpaths don't cause a collision. However, if there is a collision detected, the paths have to be somehow altered to prevent this. There are manual steps that could be taken, or, in some cases, the part geometry is impossible to print with that specific print-head. There is also the possibility of devising algorithms for collision prevention, that, in the case of a detected collision, iteratively adjust certain parameters until the collision has been averted, if possible.

[5] Power for bed heating and signal for thermistor.

6.2.4 IK and error compensation firmware integration

Section 4.5 covers the IK control of the printer movement, as well as error compensation that was applied. However, the implementation of this was done as an extra layer of software, when it would be more efficient to integrate it into the open-source firmware Marlin. The calibration for error compensation could be an automatic routine, where the printhead probes against a switch at certain angles to automatically find the deviations used for error compensation. The compensation could then be directly written to the IK equations in the firmware.

6.3 RESEARCH OBJECTIVE

Reiterating the overarching goal of the work, it was defined as demonstrating the WHY and HOW to do multi-axis FDM. Answering these two questions, I hope, can generate sufficient interest in the 3D printing community to kickoff a larger multi-axis project. To answer WHY multi-axis FDM is an attractive proposition, I investigated whether an accessible multi-axis FDM printer could address the shortcoming of a conventional one, without introducing new drawbacks. The first three research questions dealt with this portion of the objective. Then, to answer HOW multi-axis FDM can be done, I designed a simple print-head that can be retrofit to desktop 3D printers, and modified an existing G-code bending script to support multi-axis. The design embraces the principle of self-replication - the idea that 3D printers can produce themselves. Additionally, it is a compact and simple design. Ultimately this means that someone with an entry-level 3D printer can, on that printer, print most of the parts necessary to augment itself to 5 axes. They could start printing multi-axis parts using the free G-code bending technique, or some other way. Maybe they know Python and would contribute to the path-planning scripts. Maybe they know C and would contribute to firmware improvements. Maybe they just want to print some multi-axis parts, signaling to the community and industry that there is a demand for this technology.

Over the period that I've been working on this project, I've occasionally shared fragments of it with the online 3D printing community on Reddit, for public scrutiny. Following hundreds of interactions I've had with members of the community, I am confident that there is an interest in broader adoption of this technology. Even though the work I've done is rudimentary, I am eager to see where the community will take the ideas next, and I hope people will find use in the basic design that I'm putting out there.

List of References

[1] Lisa Joy and Jonathan Nolan. *Westworld*. TV series produed by HBO. 2016.

[2] Ian Gibson, David Rosen, and Brent Stucker. *Additive Manufacturing Technologies*. Second Edition. Springer, 2015.

[3] Ugur M. Dilberoglu et al. "The Role of Additive Manufacturing in the Era of Industry 4.0". In: *Procedia Manufacturing* 11 (2017), pp. 545–554. ISSN: 23519789. DOI: 10.1016/j.promfg.2017.07.148.

[4] Johannes F Gottwald. *US3596285A, Liquid metal recorder*. Patent filed by Teletype Corp. 1971. URL: https://patents.google.com/patent/US3596285A.

[5] J.-P. Kruth, M.C. Leu, and T. Nakagawa. "Progress in Additive Manufacturing and Rapid Prototyping". In: *CIRP Annals* 47 (2 1998), pp. 525–540.

[6] Flaviana Calignano et al. "Overview on additive manufacturing technologies". In: *Proceedings of the IEEE* 105 (4 2017), pp. 593–612.

[7] S. Scott Crump. *US5121329A, Apparatus and method for creating three-dimensional objects*. Patent filed by Stratasys. 1989. URL: https://patents.google.com/patent/US5121329A.

[8] Protolabs. *3D Printing Trend Report 2023*. 2023.

[9] Carolyn Schwaar. *This 3D-Printed Microturbine Engine Is Designed To Do More Than Fly*. Forbes. 2023. URL: https://www.forbes.com/sites/carolynschwaar/2023/06/08/this-3d-printed-microturbine-engine-is-designed-to-do-more-than-fly/.

[10] Kerry Stevenson. *The Very Good Reason 3D Printed Parts Were Found In The Tesla Model Y*. Fabbaloo. 2020. URL: https://www.fabbaloo.com/2020/04/the-very-good-reason-3d-printed-parts-were-found-in-the-tesla-model-y.

[11] Markus Trapp et al. "Improving Sustainability of Footwear Production through 3D Printing of Shoes". In: *Digitization of the work environment for sustainable production* (Sept. 2022), pp. 1–15. DOI: 10.30844/wgab_2022_1.

[12] *Printing the future of footwear with Zellerfeld: A design revolution is afoot*. Dissrup. 2023. URL: https://dissrup.com/editorial/printing-the-future-of-footwear-with-zellerfeld.

[13] *Interview with Martin Krona, President EMEA at Markforged*. European Security & Defence. 2023. URL: https://euro-sd.com/2023/03/articles/30052/interview-with-martin-krona-president-emea-at-markforged/.

[14] Adrian Bowyer. *Replicating Rapid Prototyper*. TEDx talk on YouTube. 2011. URL: https://youtu.be/XxJgZnhq7I4.

[15] Zondartul. *Idea: spreading the extruder traction over 4-6 gears - more nozzle pressure, less grinding*. Reddit post. 2022. URL: https://www.reddit.com/r/3Dprinting/comments/po00j0/idea_spreading_the_extruder_traction_over_46/.

[16] Filament2Print. *Direct extrusion and Bowden systems*. Forum post. 2020. URL: https://filament2print.com/gb/blog/94_bowden-direct-extrusion.html.

[17] Daniel Ahlers et al. *3D Printing of Nonplanar Layers for Smooth Surface Generation*.

[18] Raise3D. *When and How to Use 3D Printed Support Structures*. URL: https://www.raise3d.com/academy/when-and-how-to-use-3d-printed-support-structures/.

[19] Hubs. *Knowledge base: 3D printing*. URL: https://www.hubs.com/knowledge-base/3d-printing/.

[20] Yu An Jin et al. "Quantitative analysis of surface profile in fused deposition modelling". In: *Additive Manufacturing* 8 (Oct. 2015), pp. 142–148. ISSN: 22148604. DOI: 10.1016/j.addma.2015.10.001.

[21] Nima Zohdi and Richard Chunhui Yang. *Material anisotropy in additively manufactured polymers and polymer composites: A review*. Oct. 2021. DOI: 10.3390/polym13193368.

[22] Edward Gilford, Rob Thompson, and Jess Gwynne. "Teaching & Learning packages: Additive Manufacturing: Properties of FDM Prints". In: (2016). URL: https://www.poitpoms.ac.uk/tlplib/add_manuf/fdm.php.

[23] Caleb Kraft. *5 Axis 3D Printing Brings New Possibilities*. Make magazine. 2019. URL: https://makezine.com/article/maker-news/5-axis-3d-printing-brings-new-possibilities/.

[24] A. A. Yakovlev et al. "Study of the Multi-axis FFF 3D Printing Process". In: Institute of Electrical and Electronics Engineers Inc., Dec. 2020, pp. 262–266.

[25] Jingchao Jiang, Stephen T. Newman, and Ray Y. Zhong. "A review of multiple degrees of freedom for additive manufacturing machines". In: *International Journal of Computer Integrated Manufacturing* 34 (2 2021), pp. 195–211.

[26] Vladimir E. Kuznetsov et al. "Strength of PLA components fabricated with fused deposition technology using a desktop 3D printer as a function of geometrical parameters of the process". In: *Polymers* 10 (3 Mar. 2018). ISSN: 20734360.

[27] Debapriya Chakraborty, B. Aneesh Reddy, and A. Roy Choudhury. "Extruder path generation for Curved Layer Fused Deposition Modeling". In: *Computer-Aided Design* 40 (2 Feb. 2008), pp. 235–243. ISSN: 0010-4485.

[28] Donghong Ding et al. "Multi-direction slicing of STL models for robotic wire-feed additive manufacturing". In: *International Solid Freeform Fabrication Symposium 2015* (Aug. 2015), pp. 1059–1069.

[29] Frederik Wulle et al. "Workpiece and Machine Design in Additive Manufacturing for Multi-Axis Fused Deposition Modeling". In: vol. 60. Elsevier B.V., 2017, pp. 229–234.

[30] T. Vaneker et al. "Design for additive manufacturing: Framework and methodology". In: *CIRP Annals - Manufacturing Technology* 69 (2020), pp. 578–599.

[31] Wei Chen Lee, Ching Chih Wei, and Shan Chen Chung. "Development of a hybrid rapid prototyping system using low-cost fused deposition modeling and five-axis machining". In: *Journal of Materials Processing Technology* 214 (11 2014), pp. 2366–2374. ISSN: 09240136. DOI: 10.1016/j.jmatprotec.2014.05.004.

[32] Guoxin Fang et al. "Reinforced FDM: Multi-axis filament alignment with controlled anisotropic strength". In: *ACM Transactions on Graphics* 39 (6 Nov. 2020). ISSN: 15577368.

[33] Diego Garcia Cuevas and Gianluca Pugliese. *Advanced 3D printing with Grasshopper*. Independently published, 2020.

[34] Arturo Tedeschi. *AAD Algorithms-Aided Design, Parametric Strategies using Grasshopper*. First Edition. Le Penseur, 2014.

[35] T. S. Mruthyunjaya. "Kinematic structure of mechanisms revisited". In: *Mechanism and Machine Theory* 38 (4 Apr. 2003), pp. 279–320. ISSN: 0094-114X.

[36] Ekaterina Kopets et al. "Estimating Natural Frequencies of Cartesian 3D Printer Based on Kinematic Scheme". In: *Applied Sciences (Switzerland)* 12 (9 May 2022). ISSN: 20763417. DOI: 10.3390/app12094514.

[37] Achim Kampker et al. "Review on machine designs of material extrusion based additive manufacturing (AM) systems - Status-Quo and potential analysis for future AM systems". In: vol. 81. Elsevier B.V., 2019, pp. 815–819. DOI: 10.1016/j.procir.2019.03.205.

[38] Shane Hooper. "3D Printer Kinematics-CoreXY vs Cartesian". In: (2021). DOI: 10.13140/RG.2.2.13012.94088. URL: https://www.researchgate.net/publication/357163478.

[39] Ilan E. Moyer. *CoreXY: Theory*. 2012. URL: https://corexy.com/theory.html.

[40] Time Magazine. *The best inventions of 2022: 200 innovations changing how we live*. 2022. URL: https://time.com/best-inventions-2022/.

[41] Sébastien Briot and Ilian Bonev. *Are Parallel Robots More Accurate than Serial Robots?* 2007, pp. 445–456. URL: https://hal.archives-ouvertes.fr/hal-00362500.

[42] Erkan İncekar, Hüseyin Kaygisiz, and Sebahattin Babur. "Dimensional Accuracy Analysis of Samples Printed in Delta and Cartesian Kinematic Three Dimensional Printers". In: *Journal of Polytechnic* (May 2020). ISSN: 1302-0900. DOI: 10.2339/politeknik.582410.

[43] Betina Madeira Schmitt et al. "A comparative study of cartesian and delta 3d printers on producing PLA parts". In: vol. 20. Universidade Federal de Sao Carlos, 2017, pp. 883–886. DOI: 10.1590/1980-5373-mr-2016-1039.

[44] Shaik Himam Saheb, G. Satish Babu, and N. V.Subba Raju. "Relative Kinematic Analysis of Serial and Parallel Manipulators". In: vol. 455. Institute of Physics Publishing, Dec. 2018. DOI: 10.1088/1757-899X/455/1/012040.

[45] Azamat Yeshmukhametov et al. "Design and kinematics of serial/parallel hybrid robot". In: 2017, pp. 162–165. ISBN: 9781509060887.

[46] Tianyu Zhang et al. "S3-Slicer: A General Slicing Framework for Multi-Axis 3D Printing". In: *ACM Transactions on Graphics* 41 (6 Nov. 2022).

[47] Chengkai Dai et al. "Support-free volume printing by multi-axis motion". In: *ACM Transactions on Graphics* 37 (4 2018).

[48] Sitong Xiang et al. "Multi-machine tools volumetric error generalized modeling and Ethernetbased compensation technique". In: *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture* 230 (5 May 2016), pp. 870–882. ISSN: 20412975. DOI: 10.1177/0954405414564407.

[49] P Miciński, J Bryła, and A Martowicz. *Multi-axis Fused Deposition Modeling using parallel manipulator integrated with a Cartesian 3D printer*, p. 2021.

[50] Mohammed A. Isa and Ismail Lazoglu. "Five-axis additive manufacturing of freeform models through buildup of transition layers". In: *Journal of Manufacturing Systems* 50 (Jan. 2019), pp. 69–80. ISSN: 02786125. DOI: 10.1016/j.jmsy.2018.12.002.

[51] Freddie Hong et al. "Open5x: Accessible 5-axis 3D printing and conformal slicing". In: ACM, Apr. 2022, pp. 1–6.

[52] BrendonBuilds. *Ongoing development on Open5x Voron 0 printer*. Published on Twitter. 2023. URL: https://twitter.com/brendonbuilds.

[53] Panagiotis Stavropoulos et al. "Hybrid subtractive-additive manufacturing processes for high value-added metal components". In: (2020). DOI: 10.1007/s00170-020-06099-8/Published. URL: https://doi.org/10.1007/s00170-020-06099-8.

[54] Sean Mason et al. *3D Printer/CNC Mill Hybrid*. University of Calgary Engineering Design Fair. 2021. URL: https://engineeringdesignfair.ucalgary.ca/mechanical/3d-printer-cnc-mill-hybrid/.

[55] Rene K. Mueller. *3D Printing: Penta Axis (PAX) / 5 Axis Printing Option*. XYZdims. 2021. URL: https://xyzdims.com/2021/02/08/3d-printing-penta-axis-pax-5-axis-printing-option/.

[56] Michael Wüthrich et al. "A novel slicing strategy to print overhangs without support material". In: *Applied Sciences (Switzerland)* 11 (18 Sept. 2021). ISSN: 20763417. DOI: 10.3390/app11188760.

[57] Stefan Hermann. *THE ROTBOT: 4-AXIS NON-PLANAR 3D PRINTING*. CNCkitchen. 2022. URL: https://www.cnckitchen.com/blog/the-rotbot-4-axis-non-planar-3d-printing.

[58] nonplanar.xyz. *Non-planar*. GitHub repository. 2022. URL: https://github.com/FreddieHong19/Open5x.

[59] Hao Liu, Zhoupeng Liu, and Siting Hao. "Design of a throat-extended FDM extruder for multi-axis 3D printing". In: *Strojniski Vestnik/Journal of Mechanical Engineering* 67 (4 2021), pp. 180–190. ISSN: 00392480. DOI: 10.5545/sv-jme.2021.7124.

[60] Simon Vez. *The Goliath Hotend*. GitHub repository. 2023. URL: https://github.com/VzBoT3D/Goliath.

[61] Gerrit Coetzee. *Keep your nozzle hot and your prints cool*. Hackaday. 2016. URL: https://hackaday.com/2016/01/29/keep-your-nozzle-hot-and-your-prints-cool/.

[62] Ondřej Stříteský. *Does your newly assembled Original Prusa i3 MK3 print the best it can?* Prusa Research. 2019. URL: https://blog.prusa3d.com/does-your-freshly-assembled-original-prusa-i3-mk3-print-as-the-best-it-can_29445/.

[63] Shawn Moylan et al. *Proposal for a standardized test artifact for additive manufacturing machines and processes*. URL: http://www.nist.gov/el/isd/sbm/matstandaddmanu.cfm;.

[64] Fabio Alberto Cruz Sanchez et al. "Towards a standard experimental protocol for open source additive manufacturing". In: 9 (3 2014), pp. 151–167. DOI: 10.1080/17452759.2014.919553Ãŕ. URL: https://hal.univ-lorraine.fr/hal-01522630.

[65] Lara Rebaioli and Irene Fassi. "A review on benchmark artifacts for evaluating the geometrical performance of additive manufacturing processes". In: *International Journal of Advanced Manufacturing Technology* 93 (5-8 Nov. 2017), pp. 2571–2598. ISSN: 14333015. DOI: 10.1007/s00170-017-0570-0.

[66] T H C Childs. *Linear and Geometric Accuracies from Layer Manufacturing selective laser sintering laminated object manuf. fused deposition modelling*. 1994.

[67] W M Johnson et al. *Benchmarking evaluation of an open source fused deposition modeling additive manufacturing system*. 1997.

[68] Paolo Minetola, Luca Iuliano, and Giovanni Marchiandi. "Benchmarking of FDM Machines through Part Quality Using IT Grades". In: vol. 41. Elsevier B.V., 2016, pp. 1027–1032. DOI: 10.1016/j.procir.2015.12.075.

[69] Sagar Jalui et al. "Development of a Novel Test Artefact for Conformal Material Extrusion Printing Structure-Function Relationships of Biomimetic Metamaterials View project Surface Finishing of Additively Manufactured Metallic Components". In: (2022). DOI: 10.26153/tsw/44331. URL: https://www.researchgate.net/publication/367431673.

[70] Jingchao Jiang. "A novel fabrication strategy for additive manufacturing processes". In: *Journal of Cleaner Production* 272 (Nov. 2020). ISSN: 09596526. DOI: 10.1016/j.jclepro.2020.122916.

[71] Yavuz Murtezaoglu et al. "Geometry-Based Process Planning for Multi-Axis Support-Free Additive Manufacturing". In: vol. 78. Elsevier B.V., 2018, pp. 73–78. ISBN: 9781510875692. DOI: 10.1016/j.procir.2018.08.175.

[72] Chenming Wu et al. "General Support-Effective Decomposition for Multi-Directional 3-D Printing". In: *IEEE Transactions on Automation Science and Engineering* 17 (2 Apr. 2020), pp. 599–610. ISSN: 15583783. DOI: 10.1109/TASE.2019.2938219.

[73] José Luis Pérez-Castillo et al. "Curved layered fused filament fabrication: An overview". In: *Additive Manufacturing* 47 (Nov. 2021). ISSN: 22148604. DOI: 10.1016/j.addma.2021.102354.

[74] Sungwoo Lim et al. "Modelling curved-layered printing paths for fabricating large-scale construction components". In: *Additive Manufacturing* 12 (Oct. 2016), pp. 216–230. ISSN: 22148604.

[75] Daniel Ahlers. *Slic3r, non-planar thesis fork*. GitHub repository. URL: https://github.com/Zip-o-mat/Slic3r/tree/nonplanar-thesis.

[76] Michael Laws. *Achieve true 3D printing with non planar slicing*. YouTube video. URL: https://youtu.be/gmePlcUOTRw.

[77] Donghua Zhao, Weizhong Guo, and Feng Gao. "Research on Curved Layer Fused Deposition Modeling with a Variable Extruded Filament". In: *Journal of Computing and Information Science in Engineering* 20 (4 Aug. 2020). ISSN: 15309827. DOI: 10.1115/1.4045637.

[78] C.-K. Shene. *CS3621 Introduction to Computing with Geometry Notes, Surface, Basic concepts*. Michigan Technological University. 2014. URL: https://pages.mtu.edu/~shene/COURSES/cs3621/NOTES/surface/basic.html.

[79] Yamin Li et al. "Multi-Axis Support-Free Printing of Freeform Parts with Lattice Infill Structures". In: *CAD Computer Aided Design* 133 (Apr. 2021). ISSN: 00104485. DOI: 10.1016/j.cad.2020.102986.

[80] Remco Kingmans. *Anamorphic art*. URL: https://prettyniceart.com/pages/anamorphic-art.

[81] Stefan Hermann. *Non-planar 3d printing by bending g-code*. CNCkitchen. 2022. URL: https://www.cnckitchen.com/blog/non-planar-3d-printing-by-bending-g-code.

[82] Stefan Hermann. *GCodeBending*. Github repository. 2022. URL: https://github.com/CNCKitchen/GCodeBending.

[83] CNCkitchen. *Conical Slicing: A different angle of 3D printing*. YouTube video. 2023. URL: https://youtu.be/1i-1TEdByZY.

[84] Maurus Gubser. *RotBotSlicer*. Github repository. 2022. URL: https://github.com/RotBotSlicer/Transform.

[85] Heinz Löpmeier. *nozzleboss*. Github repository. 2023. URL: https://github.com/Heinz-Loepmeier/nozzleboss.

[86] Freddie Hong. *Open5x*. GitHub repository. 2022. URL: https://github.com/FreddieHong19/Open5x.

[87] *Mach4 Advanced Configuration*. Avid CNC. URL: https://www.avidcnc.com/support/instructions/software/mach4AdvancedConfiguration/.

[88] Chu A. My and Erik L.J. Bohez. "A novel differential kinematics model to compare the kinematic performances of 5-axis CNC machines". In: *International Journal of Mechanical Sciences* 163 (Nov. 2019). ISSN: 00207403. DOI: 10.1016/j.ijmecsci.2019.105117.

[89] Prusament. *Technical datasheet: Prusament PC Blend*. Prusa Polymers. 2022. URL: https://prusament.com/media/2022/10/PCBlend_Prusament_TDS_2022_16_EN.pdf.

[90] *Goniometric Stages*. Thorlabs. URL: https://www.thorlabs.com/newgrouppage9.cfm?objectgroup_id=860.

[91] Mark Carew. *OpenBuilds V-Slot Linear Rail*. Crowdsourcing campaign by OpenBuilds on Kickstarter. 2013. URL: https://www.kickstarter.com/projects/openrail/openbuilds-v-slot.

[92] Maker Store. *What is T-Slot (V-Slot) and other profiles*. URL: https://www.makerstore.com.au/blog/t-slot-v-slot-profiles/.

[93] Creality. *Ender 5-Pro*. URL: https://www.creality.com/products/ender-5-pro-3d-printer.

[94] Khaled Elashry and Ruairi Glynn. "An Approach to Automated Construction Using Adaptive Programing". In: *Robotic Fabrication in Architecture, Art and Design 2014* (2014), pp. 51–66. DOI: 10.1007/978-3-319-04663-1_4.

[95] Lost In Tech. *Extreme Vase Mode II - Extremest vase mode. How thick can you go when 3D printing vase mode?* YouTube video. 2022. URL: https://youtu.be/0DAP5Zm1jvk.

[96] Erik van der Zalm et. al. *Marlin Firmware*. 2011. URL: https://marlinfw.org/.

[97] Paul F. Jacobs and Jan Richter. "Advances in Stereolithography Accuracy". In: *1991 International Solid Freeform Fabrication Symposium* (4 1991).

[98] Vincent (Vincent6m). *Sitting cat low poly*. Thing on Thingiverse. 2018. URL: https://www.thingiverse.com/thing:3003775.

[99] Andrew Gleadall. "FullControl GCode Designer: Open-source software for unconstrained design in additive manufacturing". In: *Additive Manufacturing* 46 (Oct. 2021). ISSN: 22148604.

[100] Greg Turk. *The Stanford Bunny*. 2000. URL: https://faculty.cc.gatech.edu/~turk/bunny/bunny.html.

Appendix A

Miscellaneous

GitHub repository containing accompanying files:

https://github.com/andersonsjanis/MultiAxisThesis.git

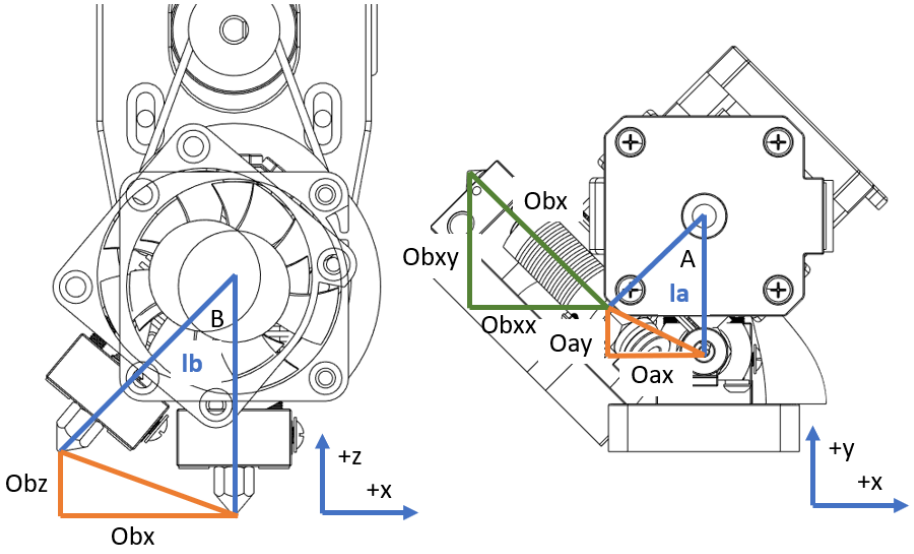
A.1 BILL OF MATERIALS

Assumes filament cost of €22.95/kg. Italicized items are harvested from host printer.

Table A.1 Bill of Materials showing breakdown of costs

Item	*Qty*	*Part Number*	*Type*	*Weight (g)*	*Unit cost*	*Total cost*	*Vendor*
1	1	Nema17 pancake stepper motor	COTS		€8.00	€8.00	TinyTronics.nl
2	1	GT2 20 T 5mm bore	COTS		€6.00	€6.00	123-3d.nl
3	1	B-driven-pulley	Print	8.1	€0.19	€0.19	
4	*1*	*Hot-end-assem*					
5	2	608 2RS bearing DIN625	COTS		€0.32	€0.64	amazon.nl
6	1	GT2 timing belt 6x158mm	COTS		€3.50	€3.50	
7	1	B_arm	Print	26.1	€0.60	€0.60	
8	1	Nema17 stepper motor	COTS		€12.00	€12.00	TinyTronics.nl
9	1	Carriage-mount	Print	26.6	€0.61	€0.61	
10	2	Microswitch	COTS		€0.68	€1.36	
11	1	B_switch_cover	Print	0.3	€0.01	€0.01	
12	1	Fan-mount	Print	2.8	€0.06	€0.06	
13	*1*	*Hot-end-fan*					
14	8	DIN912 M3x6mm sockethead bolt	Fastener				
15	8	DIN912 M3x10mm sockethead bolt	Fastener				
16	3	M3 heat-set insert	Fastener			€0.00	
17	1	Hotend_spacer	Print	0.5	€0.01	€0.01	
18	2	DIN912 M2.5x10mm sockethead bolt	Fastener				
		Total mechanical parts				€32.98	
	1	BTT Octopus board			€59.50	€59.50	TinyTronics.nl
	6	TMC2208 Stepper driver			€8.00	€48.00	TinyTronics.nl
		Total electronics				€107.50	
	1	Ender 5 Pro printer	COTS		€319.50	€319.50	123-3d.nl
		Total build cost				€459.98	

A.2 DERIVATION OF IK



From the above schematic representation we can express the true X, Y and Z values using the offsets.

$$X = X' - O_{ax} - O_{bxx} \quad \wedge \quad Y = Y' - O_{ay} - O_{bxy} \quad \wedge \quad Z = Z' - O_{bz} \tag{A.1}$$

We then express the offsets using the A and B angles. It is important to keep track of the positive directions of the coordinate systems, given in the figure.

$$O_{bz} = l_b - cosB \cdot l_b \quad \wedge \quad O_{bx} = -sinB \cdot l_b \quad \wedge \quad O_{ay} = l_a - cosA \cdot l_a \tag{A.2}$$

$$O_{ax} = -sinA \cdot l_a \quad \wedge \quad O_{bxy} = sinA \cdot O_{bx} \quad \wedge \quad O_{bxx} = -cosA \cdot O_{bx} \tag{A.3}$$

We can then substitute in the Obx expression into the Obxy and Obxx.

$$O_{bxy} = -sinA \cdot sinB \cdot l_b \quad \wedge \quad O_{bxx} = cosA \cdot sinB \cdot l_b \tag{A.4}$$

The initial X, Y and Z can then be expressed to get the IK formulation.

$$X = X' + sinA \cdot l_a + cosA \cdot sinB \cdot l_b \quad \wedge \quad Y = Y' - l_a + cosA \cdot l_a - sinA \cdot sinB \cdot l_b \quad \wedge \quad Z = Z' - l_b + cosB \cdot l_b \tag{A.5}$$

A.3 ADDITIONAL FIGURES

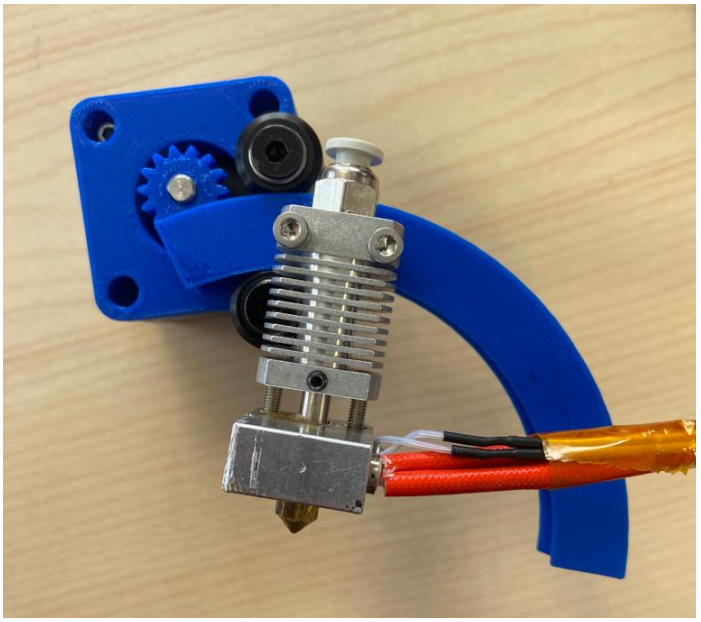


Figure A.1: Prototype B-axis assembly of VP2 prototype, showing the carrier gear which was too compliant

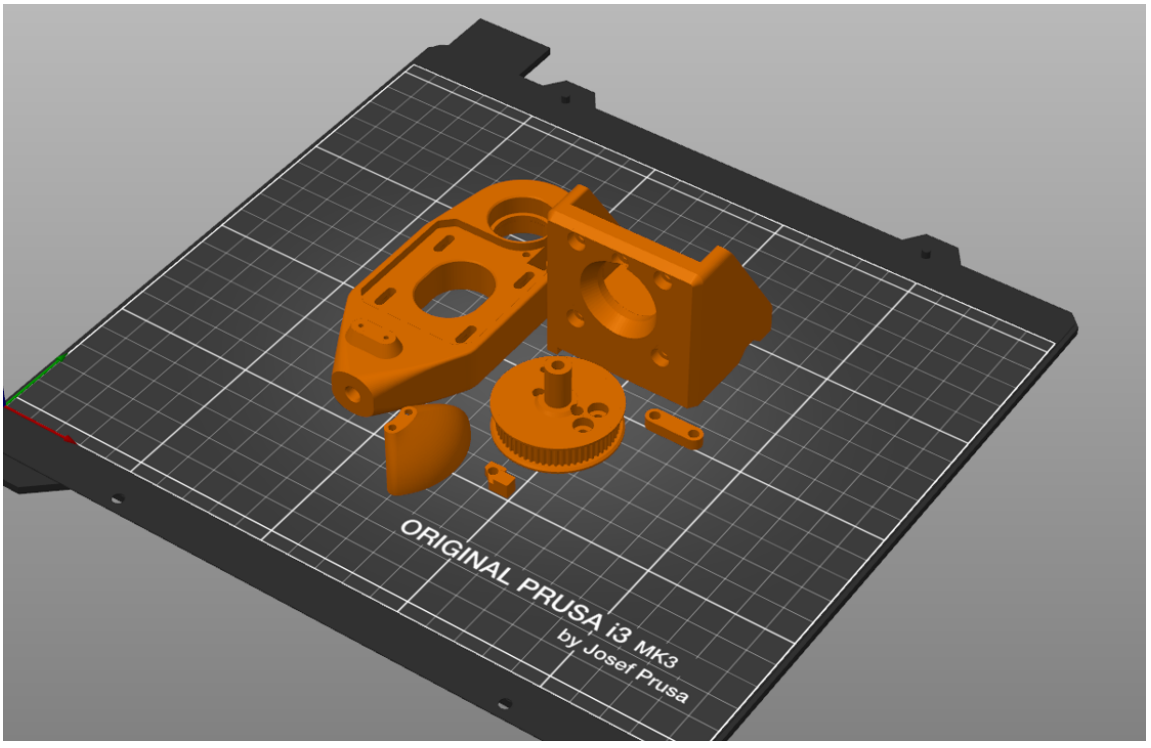


Figure A.2: Typical size (Prusa mk3) build plate shown fitting all printable parts of RW2 printhead.

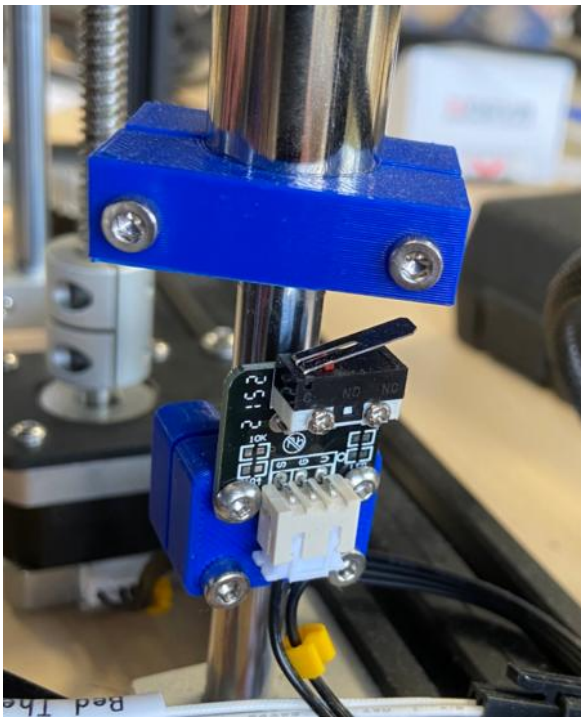


Figure A.3: Changed position of Z endswitch on Ender 5 Pro, to support RW prototypes.

Figure A.4: Planar sample part broken during support-removal

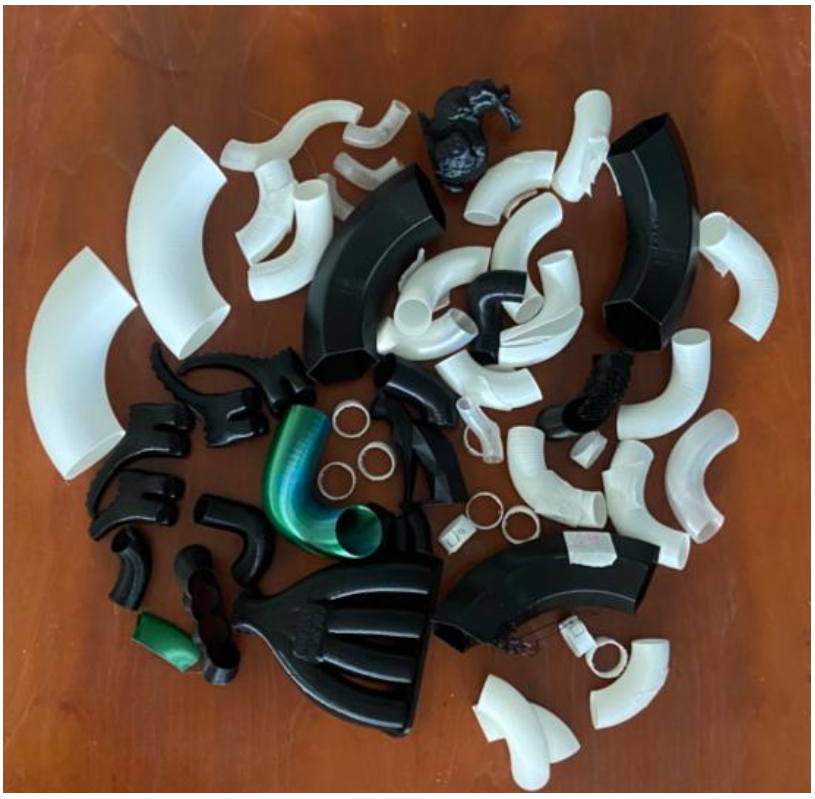


Figure A.5: Various parts printed during testing

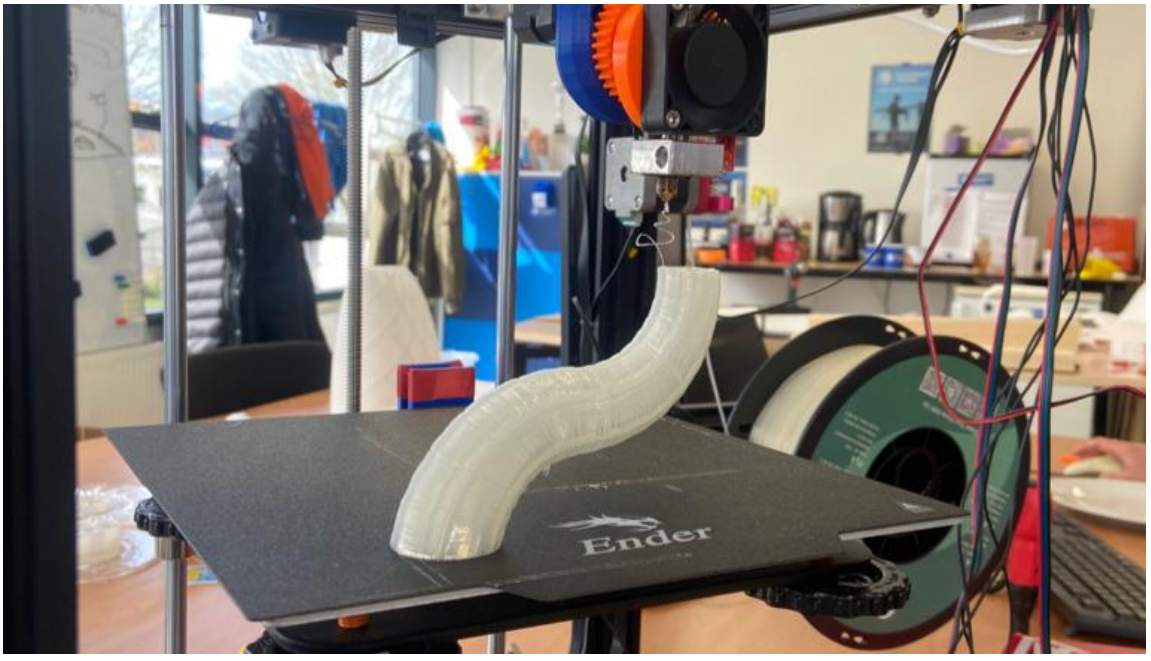


Figure A.6: S-tube printed using G-code generated with FullControl by Gleadall[99]

Figure A.7: Stanford bunny[100] being printed using G-code bending

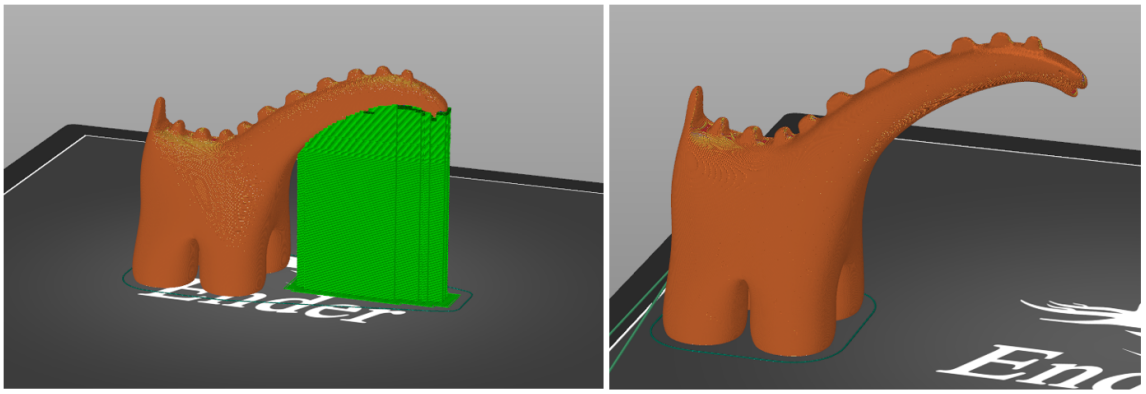


Figure A.8: Support reduction on Dinosaur model

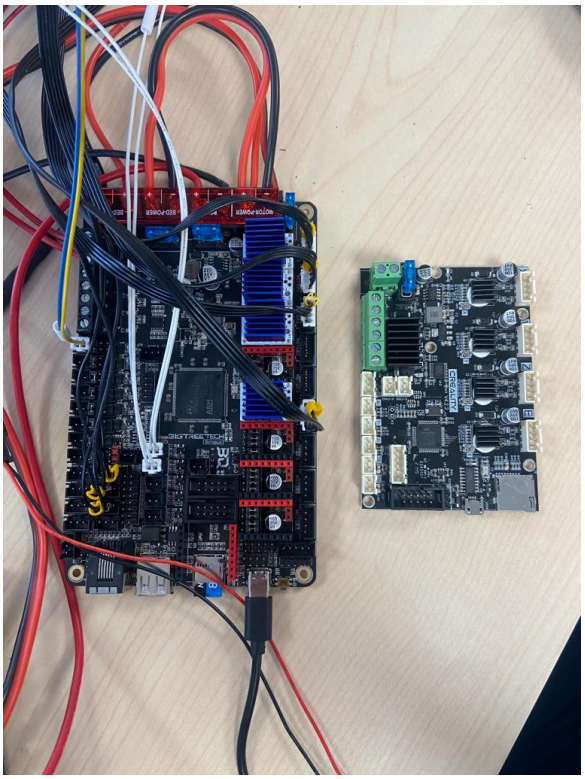


Figure A.9: BTT Octopus board, populated with TMC2208 stepper drivers, in comparison with stock Ender 5 Pro board

Appendix B

Grasshopper definitions

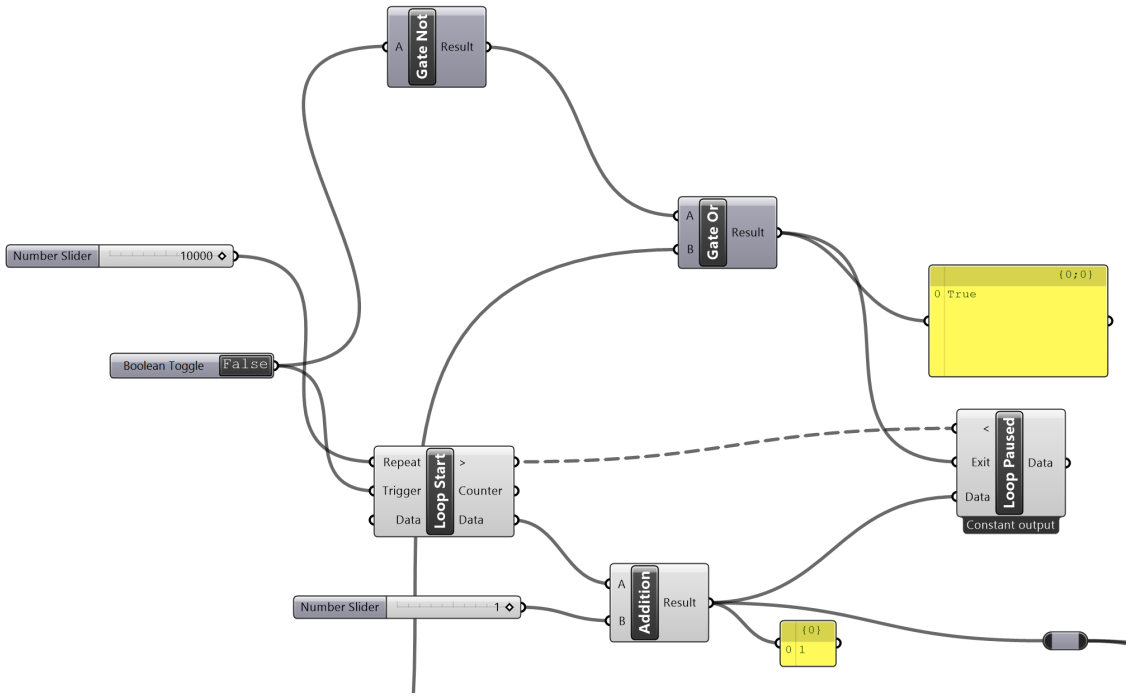


Figure B.1: Collision detection script, utilizing loop in Anemone

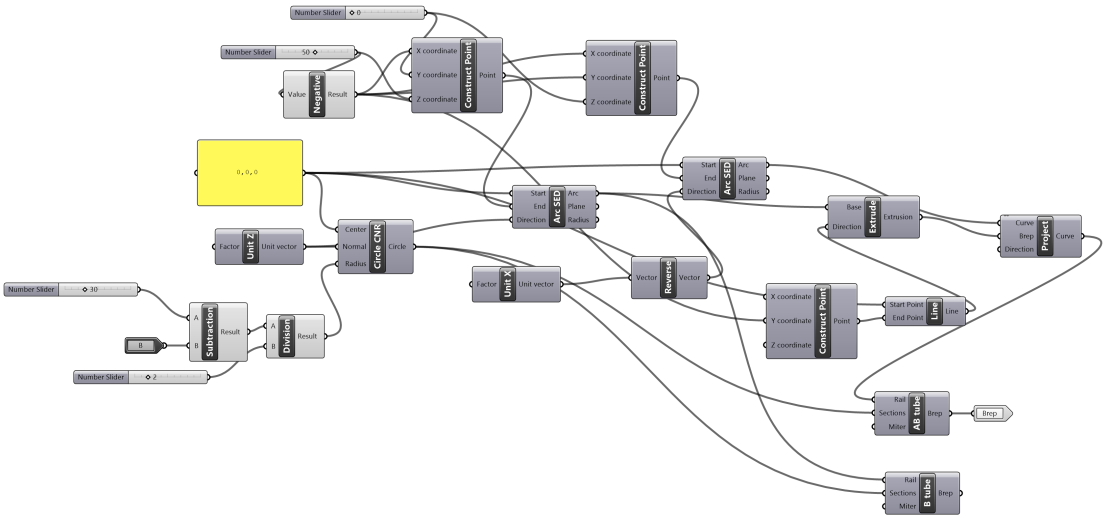


Figure B.2: Parametric model of AB-tube and B-tube sample parts

B.1 ISO-CURVES SLICING

Full GH from GitHub repository:

https://github.com/andersonsjanis/MultiAxisThesis/blob/main/isocurves_722_clustered.gh

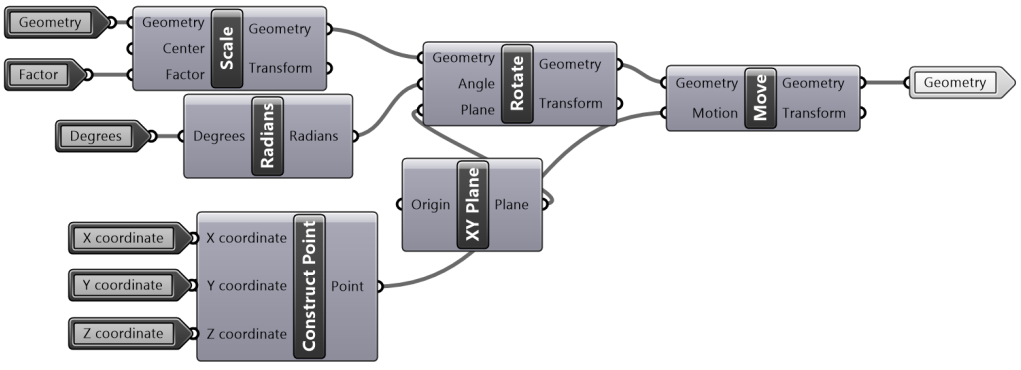


Figure B.3: Positioning cluster, which positions, orients and scales the input geometry according to the user-set parameters.

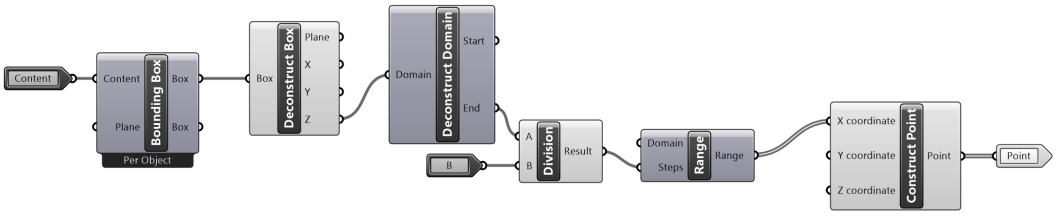


Figure B.4: Layer count cluster, determines positions of iso-curves using user-set nominal layer height

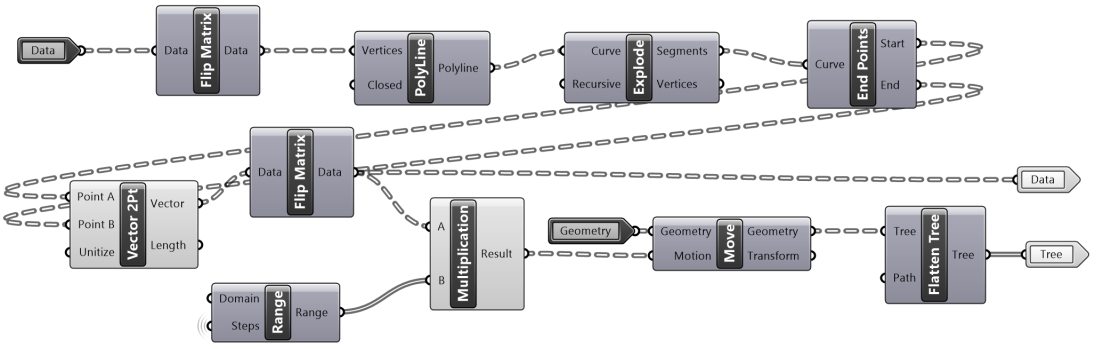


Figure B.5: Nozzle vectors cluster

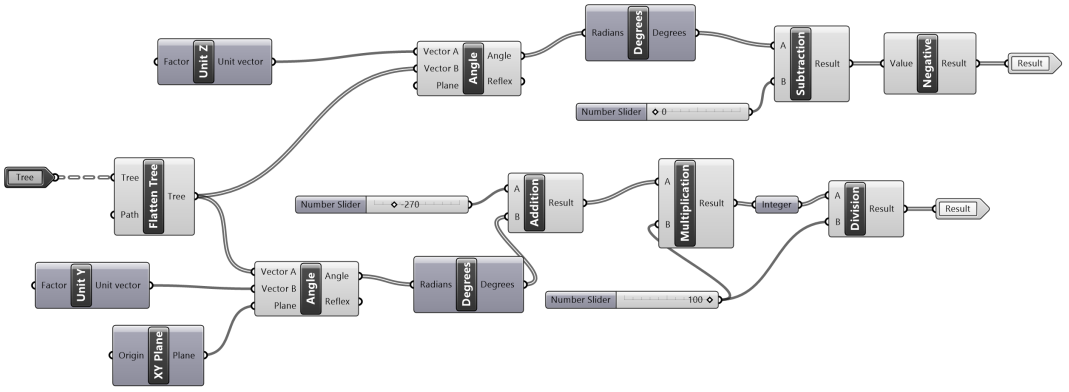


Figure B.6: Vector angle extraction cluster

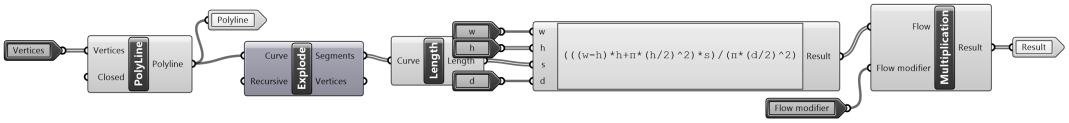


Figure B.7: Flow cluster

Appendix C

Code

C.1 START & END G-CODE ROUTINES

```
M201 X500 Y500 Z100 E1000 ; sets maximum accelerations, mm/sec^2
M203 X500 Y500 Z20 E60 ; sets maximum feedrates, mm / sec
M204 P500 R1000 T500 ; sets acceleration (P, T) and retract acceleration (R), mm/sec^2
M205 X8.00 Y8.00 Z0.40 E5.00 ; sets the jerk limits, mm/sec
M205 S0 T0 ; sets the minimum extruding and travel feed rate, mm/sec
M107
;TYPE:Custom
G90 ; use absolute coordinates
M83 ; extruder relative mode
G28 ; home all axis
G92 A-168 ; sets rotational axis positions
G1 A0 B0
M104 S150 ; set temporary nozzle temp to prevent oozing during homing
M140 S60 ; set final bed temp
G4 S30 ; allow partial nozzle warmup
G1 Z50 F240 A0 B0
G1 X2.0 Y10 F3000
M104 S220 ; set final nozzle temp
M190 S60 ; wait for bed temp to stabilize
M109 S220 ; wait for nozzle temp to stabilize
G1 Z0.28 F240
G92 E0
G1 X2.0 Y140 E10 F1500 ; prime the nozzle
G1 X2.3 Y140 F5000
G92 E0
G1 X2.3 Y10 E10 F1200 ; prime the nozzle
G92 E0
G21 ; set units to millimeters
G90 ; use absolute coordinates
M83 ; use relative distances for extrusion
G92 E0
```

tex/code/startgcode.gcode

```
;GCODE END
; Filament-specific end gcode
;END gcode for filament
G1 Z150.08 F600 ; Move print bed down
G1 X50 Y50 F9000 ; move print head out of the way
G1 Z290 F600 ; Move print bed close to the bottom
M140 S0 ; turn off heatbed
```

```
M104 S0 ; turn off temperature
M107 ; turn off fan
M84 X Y E ; disable motors
```

tex/code/endgcode.gcode

C.2 INVERSE KINEMATICS TRANSLATION LAYER

```
# -*- coding: utf-8 -*-
"""
Created on Mon Apr  3 11:38:29 2023

@author: ander
"""

import math


def recalculate(line, func_x, func_y, func_z):
    """
    Given a G-code line and three functions for X, Y, and Z,
    apply each function to its respective value in the line and
    return the modified line. If the line is not a valid G-code
    command, return it unchanged.
    """

    print("input line:", line)

    if not line.startswith("G") and not line.startswith("M"):
        return line  # Not a valid G-code command, so return it unchanged

    parts = line.split()
    x_val = y_val = z_val = a_val = b_val = 0  # Initialize all values to 0

    print("parts (before loop):", parts)

    x_indices = []
    y_indices = []
    z_indices = []

    for i in range(1, len(parts)):
        if parts[i].startswith("X"):
            x_val = float(parts[i][1:])
            x_indices.append(i)
        elif parts[i].startswith("Y"):
            y_val = float(parts[i][1:])
            y_indices.append(i)
        elif parts[i].startswith("Z"):
            z_val = float(parts[i][1:])
            z_indices.append(i)
        elif parts[i].startswith("A"):
            a_val = float(parts[i][1:])
        elif parts[i].startswith("B"):
            b_val = float(parts[i][1:])

    for i in x_indices:
        parts[i] = "X" + str(func_x(x_val, y_val, z_val, a_val, b_val))
```

```
50
    for i in y_indices:
52      parts[i] = "Y" + str(func_y(x_val, y_val, z_val, a_val, b_val))

54  for i in z_indices:
        parts[i] = "Z" + str(func_z(x_val, y_val, z_val, a_val, b_val))
56

58  print("parts (after loop):", parts)
    print("output line:", line)
60

62  return " ".join(parts)

64


66
def main(input_file, output_file, func_x, func_y, func_z):
68      """
    Given an input G-code file, an output file name, and three functions
70  for X, Y, and Z, read each line of the input file, apply the functions
    to the X, Y, and Z values, and write the modified line to the output
    file.
72      """
    with open(input_file) as f:
74      lines = f.readlines()

76  with open(output_file, "w") as f:
        for line in lines:
78          new_line = recalculate(line, func_x, func_y, func_z)
            f.write(new_line + "\n")
80

82 if __name__ == "__main__":

84


86  """
    def func_x(x_val, y_val, z_val, a_val, b_val):
88      return x_val * y_val

90  def func_y(x_val, y_val, z_val, a_val, b_val):
        return y_val*2
92
    def func_z(x_val, y_val, z_val, a_val, b_val):
94      return z_val*8
    """
96
    La = 26.5
98  Lb = 47.9

100 def func_x(x_val, y_val, z_val, a_val, b_val):
        return x_val + math.sin(math.radians(a_val)) * La + math.cos(math.
    radians(a_val)) * math.sin(math.radians(b_val)) * Lb
102
    def func_y(x_val, y_val, z_val, a_val, b_val):
104     return y_val - La + math.cos(math.radians(a_val)) * La - math.sin(
    math.radians(a_val)) * math.sin(math.radians(b_val)) * Lb

106 def func_z(x_val, y_val, z_val, a_val, b_val):
```

```
        return z_val + math.cos(math.radians(b_val)) * Lb - Lb

    input_file = "BENTexhaust2.gcode"
    output_file = "BENTexhaust2_IK.gcode"
    main(input_file, output_file, func_x, func_y, func_z)
```

tex/code/IKtranslation4.py

C.3 G-CODE BENDING

This code for G-code bending was adapted from Stefan Hermann.[81][82] This modified version uses the angleSplineThisLayer value, and outputs it as the B-axis G-code value on every line.

```
# -*- coding: utf-8 -*-
"""
Created on Wed Jan 12 10:10:14 2022
@author: stefa
Modified by Janis Andersons to output B-axis coordinates (angle)
"""

import numpy as np
import math
from scipy.interpolate import CubicSpline
import matplotlib.pyplot as plt
import re
from collections import namedtuple

Point2D = namedtuple('Point2D', 'x y')
GCodeLine = namedtuple('GCodeLine', 'x y z e f')


#################   USER INPUT PARAMETERS   ##########################

INPUT_FILE_NAME = "manifold4_1.gcode"
OUTPUT_FILE_NAME = "BENT" + INPUT_FILE_NAME
LAYER_HEIGHT = 0.28 #Layer height of the sliced gcode
WARNING_ANGLE = 100 #Maximum Angle printable with your setup

#2-point spline
SPLINE_X = [40, 130]
SPLINE_Z = [0, 130]

SPLINE = CubicSpline(SPLINE_Z, SPLINE_X, bc_type=((1, 0), (1, 2.5))) #bent
    90

DISCRETIZATION_LENGTH = 0.01 #discretization length for the spline length
    lookup table

#################   USER INPUT PARAMETERS END  ##########################


SplineLookupTable = [0.0]

nx = np.arange(0,SPLINE_Z[-1],1)

xs = np.arange(0,SPLINE_Z[-1],1)
```

```
fig, ax = plt.subplots(figsize=(6.5, 4))
ax.plot(SPLINE_X, SPLINE_Z, 'o', label='data')
ax.plot(SPLINE(xs), xs, label="S")
ax.set_xlim(0, 200)
ax.set_ylim(0, 200)
plt.gca().set_aspect('equal', adjustable='box')
# ax.legend(loc='lower left', ncol=2)
plt.show()


def getNormalPoint(currentPoint: Point2D, derivative: float, distance: float) -> Point2D: #claculates the normal of a point on the spline
    angle = np.arctan(derivative) + math.pi /2
    return Point2D(currentPoint.x + distance * np.cos(angle), currentPoint.y + distance * np.sin(angle))

def parseGCode(currentLine: str) -> GCodeLine: #parse a G-Code line
    thisLine = re.compile('(?i)^[gG][0-3](?:\s+x(?P<x>-?[0-9.]{1,15})|\s+y(?P<y>-?[0-9.]{1,15})|\s+z(?P<z>-?[0-9.]{1,15})|\s+e(?P<e>-?[0-9.]{1,15})|\s+f(?P<f>-?[0-9.]{1,15}))*')
    lineEntries = thisLine.match(currentLine)
    if lineEntries:
        return GCodeLine(lineEntries.group('x'), lineEntries.group('y'), lineEntries.group('z'), lineEntries.group('e'), lineEntries.group('f'))

def writeLine(G, X, Y, Z, A, F = None, E = None): #write a line to the output file
    outputSting = "G" + str(int(G)) + " X" + str(round(X,5)) + " Y" + str(round(Y,5)) + " Z" + str(round(Z,3)) + " A0 B" + str(round(A,3))
    if E is not None:
        outputSting = outputSting + " E" + str(round(float(E),5))
    if F is not None:
        outputSting = outputSting + " F" + str(int(float(F)))
    outputFile.write(outputSting + "\n")

def onSplineLength(Zheight) -> float: #calculates a new z height if the spline is followed
    for i in range(len(SplineLookupTable)):
        height = SplineLookupTable[i]
        if height >= Zheight:
            return i * DISCRETIZATION_LENGTH
    print("Error! Spline not defined high enough!")

def createSplineLookupTable():
    heightSteps = np.arange(DISCRETIZATION_LENGTH, SPLINE_Z[-1], DISCRETIZATION_LENGTH)
    for i in range(len(heightSteps)):
        height = heightSteps[i]
        SplineLookupTable.append(SplineLookupTable[i] + np.sqrt((SPLINE(height)-SPLINE(height-DISCRETIZATION_LENGTH))**2 + DISCRETIZATION_LENGTH**2))



lastPosition = Point2D(0, 0)
currentZ = 0.0
lastZ = 0.0
currentLayer = 0
relativeMode = False
createSplineLookupTable()
```

```
91
with open(INPUT_FILE_NAME, "r") as gcodeFile, open(OUTPUT_FILE_NAME, "w+")
    as outputFile:
93      for currentLine in gcodeFile:
            if currentLine[0] == ";":   #if NOT a comment
95              outputFile.write(currentLine)
                continue
97          if currentLine.find("G91 ") != -1:   #filter relative commands
                relativeMode = True
99              outputFile.write(currentLine)
                continue
101         if currentLine.find("G90 ") != -1:   #set absolute mode
                relativeMode = False
103             outputFile.write(currentLine)
                continue
105         if relativeMode: #if in relative mode don't do anything
                outputFile.write(currentLine)
107             continue
            currentLineCommands = parseGCode(currentLine)
109         if currentLineCommands is not None: #if current comannd is a
    valid gcode
                if currentLineCommands.z is not None: #if there is a z
    height in the command
111                 currentZ = float(currentLineCommands.z)

113             if currentLineCommands.x is None or currentLineCommands.y is
     None: #if command does not contain x and y movement it#s probably not a
     print move
                    if currentLineCommands.z is not None: #if there is only
    z movement (e.g. z-hop)
115                     outputFile.write("G91\nG1 Z" + str(currentZ-lastZ))
                        if currentLineCommands.f is not None:
117                         outputFile.write(" F" + str(currentLineCommands.
    f))
                        outputFile.write("\nG90\nM83\n")
119                     lastZ = currentZ
                        continue
121                 outputFile.write(currentLine)
                    continue
123             currentPosition = Point2D(float(currentLineCommands.x),
    float(currentLineCommands.y))
                midpointX = lastPosition.x + (currentPosition.x -
    lastPosition.x) / 2  #look for midpoint
125
                distToSpline = midpointX - SPLINE_X[0]
127
                #Correct the z-height if the spline gets followed
129             correctedZHeight = onSplineLength(currentZ)

131             angleSplineThisLayer = np.arctan(SPLINE(correctedZHeight, 1)
    ) #inclination angle this layer

133             angleLastLayer = np.arctan(SPLINE(correctedZHeight -
    LAYER_HEIGHT, 1)) # inclination angle previous layer

135             heightDifference = np.sin(angleSplineThisLayer -
    angleLastLayer) * distToSpline * -1 # layer height difference

137             transformedGCode = getNormalPoint(Point2D(correctedZHeight,
    SPLINE(correctedZHeight)), SPLINE(correctedZHeight, 1), currentPosition.
```

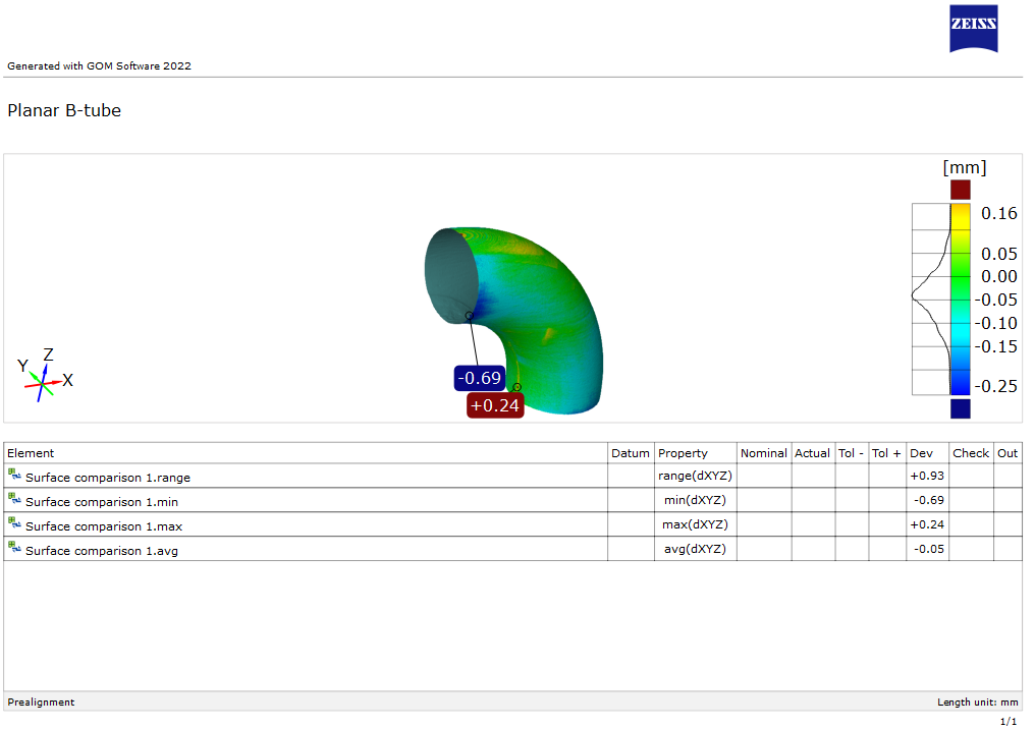
```
    x - SPLINE_X[0])

139                 #Check if a move is below Z = 0
                    if float(transformedGCode.x) <= 0.0:
141                     print("Warning! Movement below build platform. Check
    your spline!")

143                 #Detect unplausible moves
                    if transformedGCode.x < 0 or np.abs(transformedGCode.x -
    currentZ) > 50:
145                     print("Warning! Possibly unplausible move detected on
    height " + str(currentZ) + " mm!")
                        outputFile.write(currentLine)
147                     continue
                    #Check for self intersection
149                 if (LAYER_HEIGHT + heightDifference) < 0:
                        print("ERROR! Self intersection on height " + str(
    currentZ) + " mm! Check your spline!")
151
                    #Check the angle of the printed layer and warn if it's above
     the machine limit
153                 if angleSplineThisLayer > (WARNING_ANGLE * np.pi / 180.):
                        print("Warning! Spline angle is", (angleSplineThisLayer
    * 180. / np.pi), "at height  ", str(currentZ), " mm! Check your spline!"
    )
155
                    if currentLineCommands.e is not None: #if this is a line
    with extrusion
157                     """if float(currentLineCommands.e) < 0.0:
                            print("Retraction")"""
159                     extrusionAmount = float(currentLineCommands.e) * ((
    LAYER_HEIGHT + heightDifference)/LAYER_HEIGHT)
                        #outputFile.write(";was" + currentLineCommands.e + " is"
     + str(extrusionAmount) + " diff" + str(int(((LAYER_HEIGHT +
    heightDifference)/LAYER_HEIGHT)*100)) + "\n")
161                 else:
                        extrusionAmount = None
163                 Baxis = angleSplineThisLayer * 57.2958
                    writeLine(1,transformedGCode.y, currentPosition.y,
    transformedGCode.x, Baxis, None, extrusionAmount)
165                 lastPosition = currentPosition
                    lastZ = currentZ
167             else:
                    outputFile.write(currentLine)
169 print("GCode bending finished!")
```

tex/code/bend_gcode.py

Appendix D

Characterization

D.1 DIMENSIONAL ACCURACY



Element	Datum	Property	Nominal	Actual	Tol -	Tol +	Dev	Check	Out
Surface comparison 1.range		range(dXYZ)					+0.93		
Surface comparison 1.min		min(dXYZ)					-0.69		
Surface comparison 1.max		max(dXYZ)					+0.24		
Surface comparison 1.avg		avg(dXYZ)					-0.05		

Prealignment — Length unit: mm

1/1

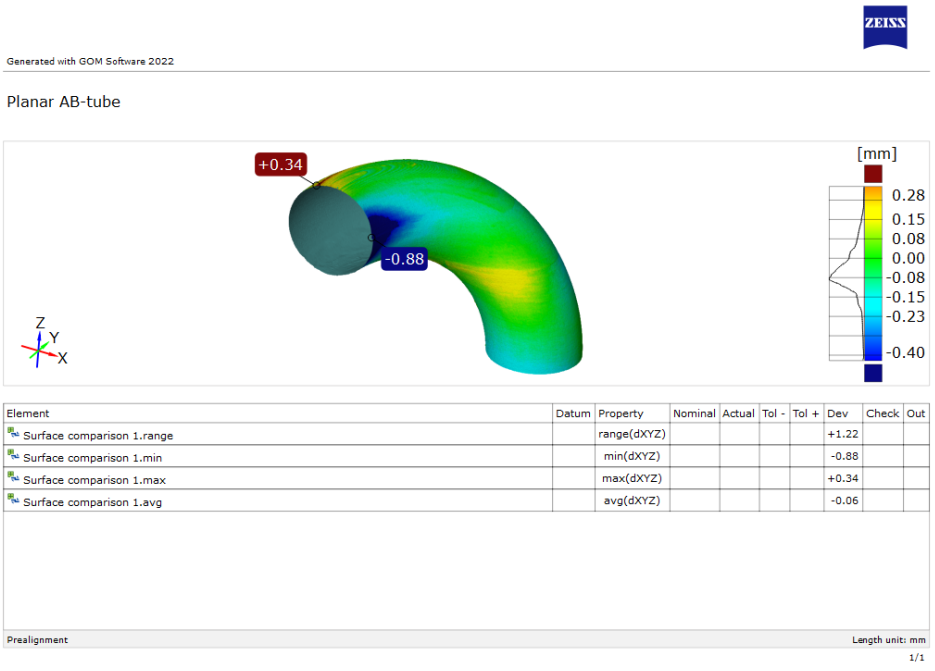
Figure D.1: Surface comparison report from GOM for planar B-tube

Element	Datum	Property	Nominal	Actual	Tol -	Tol +	Dev	Check	Out
Surface comparison 1.range		range(dXYZ)					+0.78		
Surface comparison 1.min		min(dXYZ)					-0.48		
Surface comparison 1.max		max(dXYZ)					+0.29		
Surface comparison 1.avg		avg(dXYZ)					-0.14		

Prealignment Length unit: mm

1/1

Figure D.2: Surface comparison report from GOM for non-planar B-tube



Element	Datum	Property	Nominal	Actual	Tol -	Tol +	Dev	Check	Out
Surface comparison 1.range		range(dXYZ)					+1.22		
Surface comparison 1.min		min(dXYZ)					-0.88		
Surface comparison 1.max		max(dXYZ)					+0.34		
Surface comparison 1.avg		avg(dXYZ)					-0.06		

Prealignment Length unit: mm

1/1

Figure D.3: Surface comparison report from GOM for planar AB-tube

Figure D.4: Surface comparison report from GOM for non-planar AB-tube

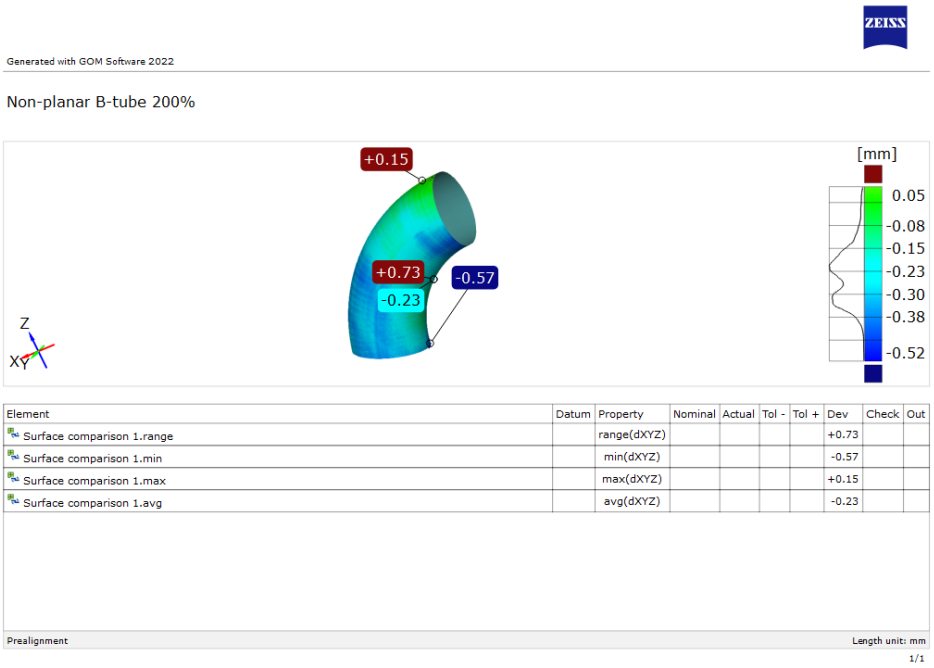


Figure D.5: Surface comparison report from GOM for 200% scaled non-planar B-tube

D.2 SURFACE QUALITY

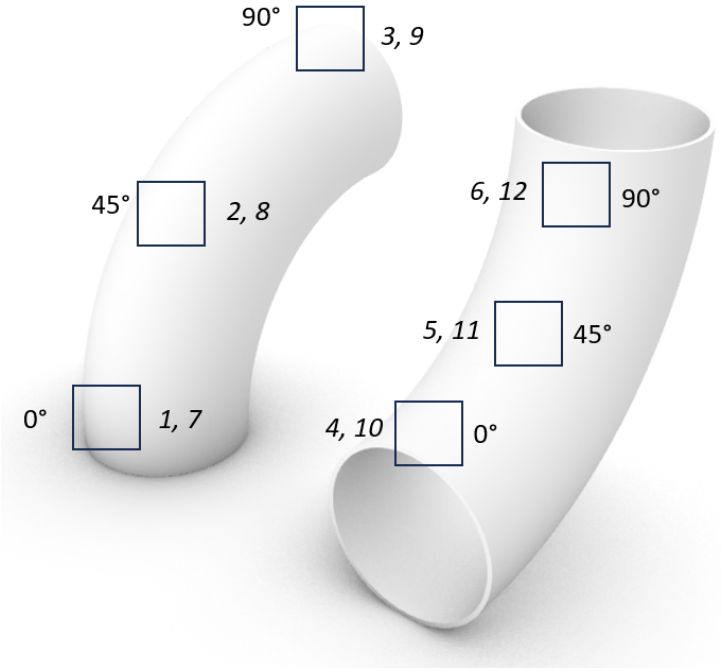


Figure D.6: Surface sampling areas. 1-6 planar, 7-12 non-planar.

Reports in GitHub repository:

https://github.com/andersonsjanis/MultiAxisThesis/tree/main/Characterization