AUTOMATED FREE-FORM FABRICATION

Developing a more accessible Non-planar workflow for the AEC industry.

C. CHIDIAC,
UNSW, Sydney, Australia
CJC_99@hotmail.com

Abstract. Three-Dimensional (3D) Printing, is a system for which 3D models, created in Computer-Aided Design (CAD) programs, are printed as tangible objects. Fused Deposition Modelling or FDM is one strategy of 3D printing where an object is constructed when the extruder selectively deposits melted material in a predetermined path in 2D layers through an extruder. This process has constant approximation errors with rounded geometries referred to as stair-stepping. This issue is prevalent throughout the 3D printing realm appearing in almost all 3D printing forums. This artefact can be lowered, by using smaller a layer height/thickness or with manual post-processing, but never indeed removed. However, to automate this process and remove the stair-stepping without any physical labour, Curved Layer Fused Deposition Modelling (CLFDM), or Non-planar printing (Chakraborty et al., 2008, pg. 235–243) can be used. This process uses all three axes of commercially available 3D printer (X, Y, Z), simultaneously, to produce stronger and smoother looking parts with less material use (Llewellyn-Jones et al., 2016, pg. 236–243). This method uses curved/non-planar layers to follow the surface of the geometry instead of spreading it over the differing layers creating the stair-stepping effect (D. Ahlers 2018, pg. 3). While current strategies exist, the process is complex. This scheme requires a Linux operation system, proficiency in the coding language C++, advanced understanding of G-code and its processes, and manipulation of a free-to-use slicer software (Ahlers, D. 2019, pg. 23–62). All of this, to have access to the settings which allow for non-planar printing to be possible. This project supplies an approachable alternative. Providing access to a broader range of individuals in the AEC industry with little to no computational knowledge — using the Rhino/Grasshopper software. Thus, higher-quality prints can be manufactured faster, stronger, and with less material waste than using traditional planar printing.

Keywords. 3D Printing, Computer-Aided Design (CAD), Fused Deposition Modelling (FDM), Curved Layer Fused Deposition Modelling (CLFDM), Non-Planar printing, G-Code, Rhino/Grasshopper.

1. Introduction: (Research context and motivations)

One of the biggest issues that plagues the 3D printing industry, particularly FDM printing, are visible layer lines. Also known as the stair-stepping artefact, these layer shifts take a magnificently complex shape, and lower it to the level of a rough draft. The FDM process, due to the large nozzle and layer heights, has constant approximation errors with rounded geometries.

Figure 1.1 A quarter sphere with generated layers (green) indicating proposed print material. The visible stair-stepping artefacts lead to approximation errors from the model (red).

As shown in *Figure 1.1* this can be quite an issue for makers looking to 3D printing to create freeform objects. This approximation error can be lowered, by using a smaller layer height/thickness or with manual post-processing methods, but never indeed removed. Searching through any 3D printing website or forum, will highlight how significant this post processing is for a cleaner finish to your prints. Methods such as sanding and priming, or acetone baths are of the most common fixes for this issue.

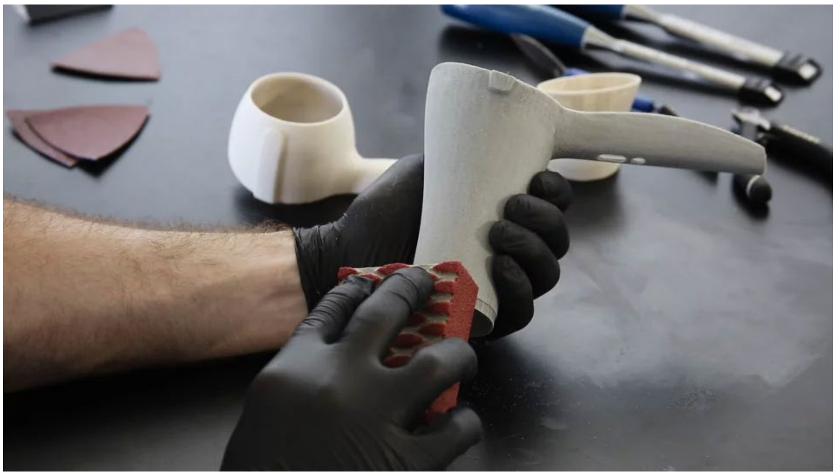

Figure 1.2 Example of the sanding and priming sequence after 3D printing.

However, they can be both physically demanding and very expensive for the average maker. Thus, a new system is needed to automate this process, without being outside the understanding of a person without computational knowledge.

Non-planar or multi-axis printing is a process by which 3D prints are printed using the X, Y and Z axis all at the same time. This process is different from that of standard printing strategies where the Z-axis is delegated to only upwards movement as the print head passes through the vertical layers that have already been printed on. Non-planar printing is a very new and emerging strategy for the construction of 3D prints, with many questions that are yet to be answered. The intended purpose of using the Non-planar printing system, over more traditional additive methods, is its ability to make printing faster, more durable, and more sustainable as, it will require less printing material than the step-style printing system that has quickly become commonplace. This new strategy hopes to encourage prototyping throughout a multitude of industries and fixes some of the fragility issues of step-style printing.

Figure 1.3 A comparison of Planar (Left) and Non-planar (Right) layers.

While a system does currently exist for this non-planar printing, the workflow is quite complex. The user's computer must contain an up to date version of Linux, for the current system to even function. Next, an intense understanding of coding languages is required to allow Linux to understand some of the factors that allow non-planar printing to work. And finally, download and change the internal code of one specific free to use slicer to have access to the predefined settings. Hours of work in order to view and access the setting.

My project will simplify the process by which this printing strategy can be achieved. This new workflow will remove this complexity and allow newcomers to understand how the entire process works without extensive knowledge of complex ideas. Additionally, I will also be working on fixing many of the issues that may arise when using the non-planar printing method, as the tool head moves through layers that have already been printed on, which can cause excess collisions with printed objects, like support structures and other printed pieces. This research project is quite significant to the 3D printing domain as it will provide a user-friendly alternative for producing higher quality prints without requiring a complex computational understanding.

2. Research Aims

The goal of this work is to develop an alternative method to print nonplanar layers on top of planar layers in any geometry. As such, this will lead to smoother surfaces with less stair-stepping artifacts. The toolpaths that are generated will be printable completely functional on any entry-level three-axis 3D printer without the need to further modify the machine. The printing strategy will provide faster and smoother results than traditional planar layers. The surfaces are identified by their angle and will be checked for collisions during the print process. The toolpaths are generated for a planar layer and then contours will be applied along the original surface mesh. The software used will allow for the script to be made public for a more ease of access. Food4Rhino will play host to this plugin.

3. Research Question(s)

Based on the issues outlined in the introduction and the derived aims, the question this research project investigates is:

How can a Non-planar workflow can be made more accessible to individuals in the AEC industry with limited computational understanding? And how are Rhino/Grasshopper able to correct issues that exist in the current non-planar workflow?

4. Methodology

Action research, or AR, is the process by which one seeks transformative change through concurrent action and research. By doing this, one is able to critically analyse and reflect on the project at hand. The methods involved in AR, have the potential to produce highly relevant research results and allow for the unification of both the researcher and the practitioner in a clean and efficient manner. A manner for which will be used, and worked from, to accomplish the aims of this research paper. A steady iterative design is the basis for which this project has been conducted where an intuitive non-planar workflow becomes the final artefact.

Richard L. Baskerville's investigation into information systems with Action Research highlighted the notion that when the researcher intervenes, the researcher becomes part of the study, i.e. One of the study subjects. Therefore, AR's empirical nature incorporates interpretive statements that include the observer's values and prior knowledge onto the observations. This research project clearly emphasises this, as through the owning and use of an FDM printer, I have had ample experience with refining concepts that could successfully be considered printable. And with this all the experience that comes with the countless failures. As such, AR, in this context, has allowed for the perceived "meaning" of the observation by myself as the researcher. While the researcher attempts to understand what is being observed, their personal understanding will inevitably bleed into the recording of the observation and the inferences that follow (Kant, 1908).

Baskerville and Kant's observations of the ways in which AR operates, and the researcher's involvement with the results are very pertinent to the research project I am pursuing. Thus, with my considerable understanding of the mechanisms which drive 3D printing, this information has bled into many of my assumptions of what are possible on an entry level FDM (Fused Deposition Modelling) machine.

Iterations based on what will work in the allotted timeframe, are developed throughout this project, where an emphasis is placed on the non-planar layers first, before the massing of the simple planar layers. In conjunction with this, an understanding of the ways in which Rhino/Grasshopper function has allowed me to pursue avenues with the most chance of success, when developing the smoother non-planar layers that accompany the 3D objects. Thus, the workflow developed throughout this paper will show a clear shift to allow this script to be used by a broader audience, without the need for any complex computational understanding.

5. Background Research/Literature review

5.1. 3D PRINTING BASICS

Three-Dimensional (3D) Printing, is a system for which 3D models, typically created in Computer-Aided Design (CAD) programs, can be physically printed as tangible objects. As such, this system opens the way for much more complex forms of geometry to be constructed, which would otherwise be impossible to create via traditional fabrication methods. And, with the new technology, expiring patents allowed them to become much more widespread to the mainstream consumer.

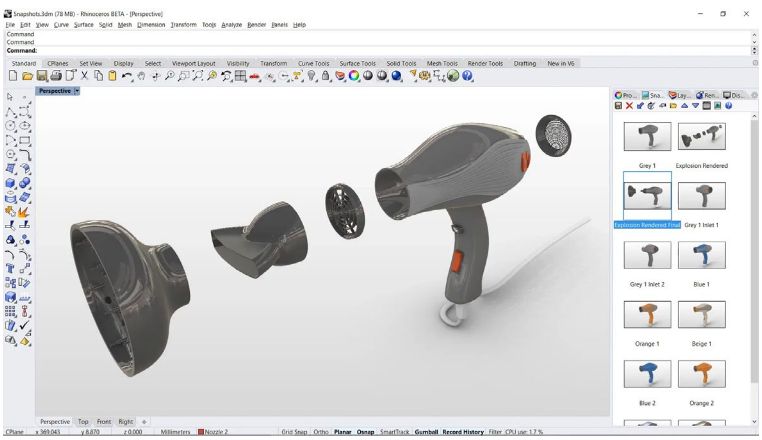

Figure 5.1. Rhinoceros 6, a CAD program, that can be used to facilitate a 3D printable model.

A very common material that has become more widespread, is the use of plastics. On top of it being a cheaper material, plastics such as ABS and PLA (two of the most commonly used plastic filaments) have an excellent melting point of anywhere between 185-225°C. This with its high strength to weight ratio and impact resistance, once cooled, supplies a versatile material that can be used as either a prototype or even as a finished product. Ceramics, clay, and metals are also used in this process, however, requiring specialty equipment to print.

5.2. PRINTING METHODS

Multiple methods of additive 3D printing are currently used today in commercial products: SLA, SLS and (the one I'll be mostly focusing on) FDM. Stereolithography, or more commonly SLA, is a technique that employs a high-powered laser to harden or 'cure' liquid resin contained within the printing reservoir to slowly build the 3D object. Selective Laser Sintering or SLS is an additive manufacturing technique that also uses a laser to 'sinter' (heating to form one solid object) a powdered material into the desired shape. Neither of these processes is very useful at producing the non-planar layers required for this system, so the third most common method must be utilized. Fused Deposition Modelling or FDM, is a system that involves the extrusion of material, in most cases plastic, from a point from which material may be deposited, an extruder. In FDM, an object is constructed when the extruder selectively deposits melted material in a predetermined path, layer by layer.

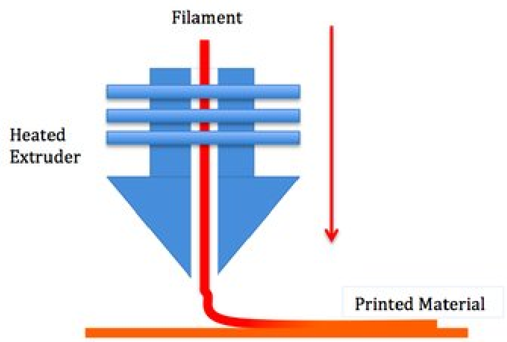

Figure 5.2. Exploded view of the mechanisms that drive FDM printing.

Much like an ordinary 2D printer, this process works from the ground up, printing its first layer on the printers' base plate, and slowly adding more and more material above the last, until the desired object is created. This system, unlike the prior processes, requires no laser and can move in the X, Y and Z axis of space.

5.3. NON-PLANAR PRINTING

Non-planar printing is the process by which a printer uses all three axes of a commercially available 3D printer (X, Y, Z), at the same time, to produce stronger and smoother looking parts with less overall material use. The stair-stepping artefact, produced by standard planar prints, reduces the surface quality of 3D printed objects as well as its structural integrity in the direction the print was oriented. It is this flaw which the non-planar printing strategy fixes. Its more dynamic and free-flowing movement paths improve 3D prints both visually and structurally as its movement strategies allow for a more optimal adhering of layers. The combination of these two printing strategies allows for smoother and stronger part production while still being printable on conventional FDM printers. Being a very new and emerging way of fabricating 3D objects, non-planar prints can be quite a complex endeavour to eventually produce.

Earlier researchers on this topic have paved the way for this new form of printing and have explored how non-planar surface generation can easily reduce this artefact. However, supplying a complex system for which the average user would have no understanding of how to operate.

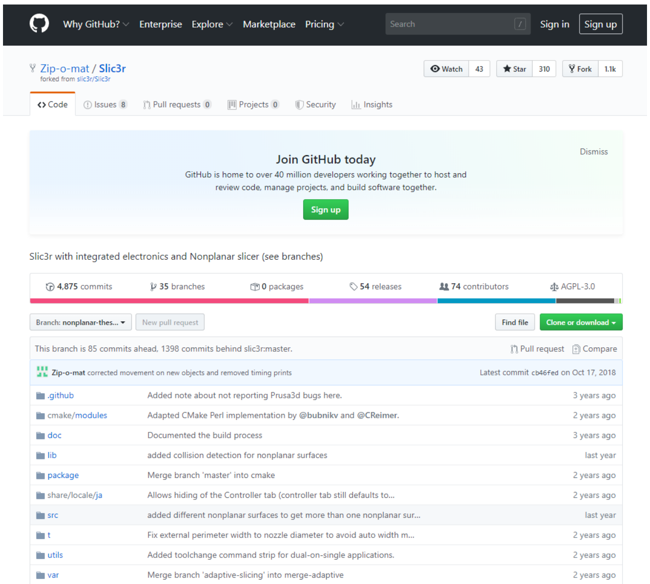

Figure 5.3. The GitHub site that currently hosts one of the only non-planar methods that is open source (Daniel Ahlers, 2018).

This research project tries to simplify this process, and subsequently allows this strategy to be more accessible to a broader range of individuals who do not have advanced computational knowledge. The software used will generate the necessary tool head pathway that will allow non-planar printing to be possible. During the fabrication stage, the printer nozzle will not collide with the object as this will all be preprogramed into the final file to check for any collisions.

5.4. ISSUES WITH FDM

With FDM as its basis, no medium of fluid or powder is needed for this process to function. This printing strategy provides a perfectly basis for non-planar printing, as this X, Y, Z movement system can be taken advantage of, to allow the print head free movement through layers that have already been printed on.

Unfortunately, the FDM process has constant approximation errors with rounded geometries, causing the stair-stepping artefact. This approximation error can be lowered by using a smaller layer height/thickness or through post processing such as sanding and priming, and acetone baths.

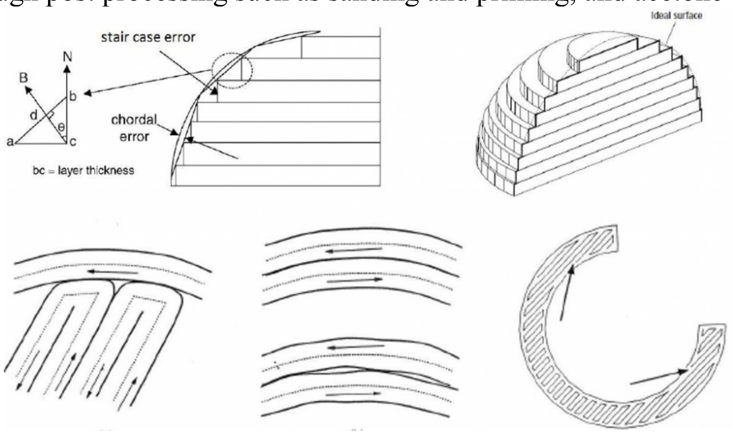

Figure 5.4. The stair-stepping artefact that occurs when attempting to create rounded geometries.

These, however, never truly remove these layer lines; only lessoning their appearance while weakening the object.

5.4. MIXING STRATAGIES

To automate this process and remove the stair-stepping without and physical labour, D. Chakraborty, B. Aneesh Reedy, and A. Roy Choudhury, have proposed a new form of FDM for the surface of the object. Curved Layer Fused Deposition Modelling, CLFDM. Further realized by Daniel Ahlers, this method uses curved/non-planar layers, to follow the surface of the geometry instead of spreading it over the differing layers creating the stair-stepping effect. The manipulation of the digital objects G-code (the coding language that determines the X, Y, Z coordinates and the deposition quantity of the print head when printing) is was needed for this process to function optimally. Supplementary research into the topic by Huang and Singamneni and Llewellyn-Jones noted that the combination of the two processes would allow for the layers to be built where the planar structure sits inside of the non-planar surface.

6. Case Study

6.1. RESEARCH

Research was among the most crucial aspects of this project, as at first, I had little to no understanding of what non-planar printing was outside of the smoothness of its surface layers. As such, for 4-5 weeks of this research task, it was imperative that I understood the mechanisms behind 3D printing, what the best style of printing was, and what software could be easily manipulated to allow for more accessibility to a broader range of people.

What I understood and became critical to completing this task are as follows:

3D printing can be separated into three main parts: Model generation, Slicing, and Printing. In model generation, the model is design, downloaded or scanned into a virtual 3D environment. Slicing refers to the printing instructions that are generated, from this virtual model, so that a 3D printer can understand the paths it must take to print the geometry. And finally, the printing of the object itself. To be printable, the STL model must be watertight and manifold else issue with missing segments will occur. Watertight, in this instance, refers to the digital model having no holes or small gaps in its surface. Many CAD programs, like Rhino/Grasshopper, can output STL models. STL files can be stored in two ways. The first is American Standard Code for Information Interchange (ASCII) or binary format. In the ASCII format, all coordinates are stored as written float values with a describing text, however, this system is quite inefficient and can take up a lot of storage space.

$$
\begin{array}{l}
\textbf{solid } \mathit{name} \\
\left\{ \begin{array}{l}
\textbf{facet normal } n_i \; n_j \; n_k \\
\quad \textbf{outer loop} \\
\qquad \textbf{vertex } v1_x \; v1_y \; v1_z \\
\qquad \textbf{vertex } v2_x \; v2_y \; v2_z \\
\qquad \textbf{vertex } v3_x \; v3_y \; v3_z \\
\quad \textbf{endloop} \\
\textbf{endfacet}
\end{array} \right\}^{+} \\
\textbf{endsolid } \mathit{name}
\end{array}
$$

Figure 6.1. ASCII STL file format. Each triangle, that makes up the mesh, is stored with the coordinates of its three vertices and a facet normal (faces orientation).

Often, the Binary format is used providing close to 80% shrinkage in file size. G-code is the numerical control language, used in CAM, to allow for the use of automated machining tools like CNC mills, laser cutters and 3D printers. G-code tells the machine how to move its axis, how fast and what locations to move to, in order to create the intended object. G or M suffix the string, G referring to introductory commands and M for miscellaneous functions. Examples of G-code include:

G1 X50 Y100 Z1.5 E10 F2400 Moves linear print head to the position of x 50, y 100, z 1.5 on the print bed and moves the extrusion motor 10 mm clockwise with the maximum speed of axis movement being 2400 mm/min.

This is the essence of 3D printing. The example script above is a single line of code in long list of commands that are sent to the printer. Varying the values produced by the axis to create non-planar layers, that morph across the geometries requires a software that stores the coordinates of the lines from start to finish.

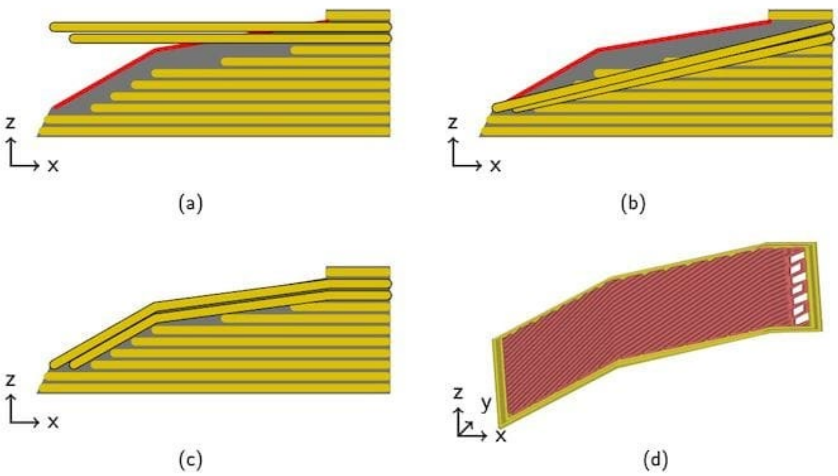

Figure 6.2. A section of a typical print path showing (a) the un-projected planar toolpaths extended out in line with its end goal, (b) the points are projected downwards to its end goal, (c) the line intersections are referenced and corrected, and (d) an example toolpath of the top most layer.

6.2. UNDERSTANDING G-CODE IN GRASSHOPPER

Acting as the slicer software, in conjunction with Simplify3D, for this project, it was my job to understand how Rhino/Grasshopper understood geometries and how to break them down into line paths for printing. To create the final G-Code that could be used for printing, the digital model must be broken down into lines that correspond to the chosen layer height.

Rhino/Grasshopper, acting as the primary slicer, takes the 3D geometry and translates this model into individual two-dimensional layers. The X, Y, Z coordinates along each line are recorded and translated into G-code that the printer will use for printing. In general, these corresponding layers are made up of two primary parts: the perimeters and infill. The perimeters are what form the hard-outer shell of the object, while the infill is the pattern that allows the part to have structure and rigidity while also saving material. In this way, the plugins Droid and Xylinus had been used to automate this layering process, generating a virtual simulation of the print heads toolpath and subsequently generating the final G-code.

6.3. ITERATIONS

Throughout the entirety of this project, the iterative process that was developed, had mostly been digital as Rhino/Grasshopper provided a visual understanding of problems that would arise when transferred to the printer. Thus, many of the mistakes could be found and resolved quicker than having to wait for prints to be done. However, there were some issues that Rhino/Grasshopper could not identify immediately, like tool path movement outside of the printing process, and required physical testing to identify and fix these issues.

6.3.1. Iteration 1 – Contours

The first iteration, and probably the most important, was establishing what method I would use to generate these curved tool paths of any shape placed into Rhino/Grasshopper. One system that came to mind was creating an alternating line system that would be projected onto the top of any model. This proved inefficient as it would miss many of the smaller details of the model. On top of this, pulling the angle from the lines to create a collision detection system proved to be quite difficult, so a new strategy had to be formulated.

A component which is provided by rhino/Grasshopper, that worked perfectly for this, providing a clean projection onto the parts and vector values, was contours. With this component, any shape can have a line system mapped onto its surface, while also providing the angle at each point on the line segment. These contours would become the basis for which I would generate the non-planar layers for the rest of this project.

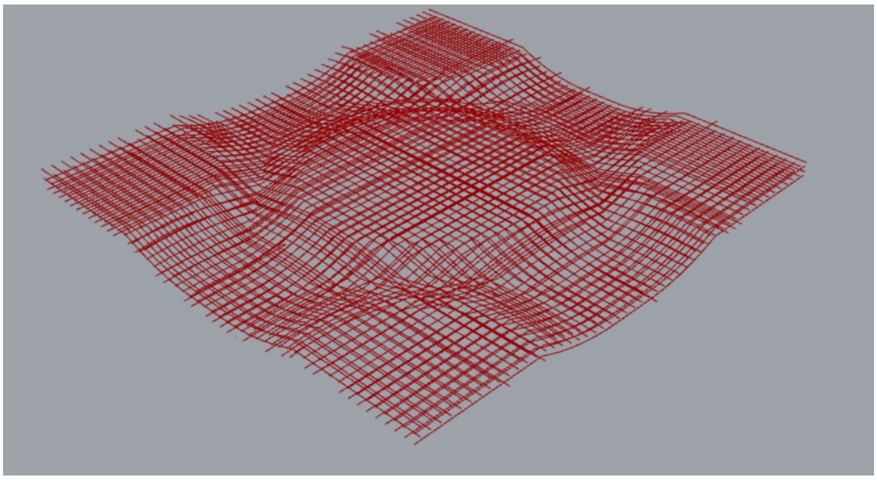

Figure 6.3.1. Iteration 1 with the use of contours as the non-planar layers, in the cross hatched pattern, using Rhino 6/Grasshopper.

6.3.2. Iteration 2 – Hollow models

Iteration 2 of my non-planar workflow resolved many of the issues of the initial design. Refining the line generation process that the contours supplied had helped in generating consistent line work that perfectly mapped to the 3D model. On top of this, some editing of the script allowed for the lines to be applied into separate directions and then sequenced based on how many top layers the individual required. Once a system resembling non-planar layers was done, the next step was to generate the raw planar layers that would act as the massing for which the non-planar layers could be applied. Looking into plugins that could automate this process, Droid was one that seemed the most useful. This plugin allowed for a quick and easy generation of planar layers, and allowed for the addition of other line types, rather than the static two-dimensional layering. One issue that did arise, however, was the model being printed hollowed out. Droid provided the outer planar lines but completely removed the bottom surface and infill of the piece. After some work, this was rectified by making sure the model was placed flush with the XY axis.

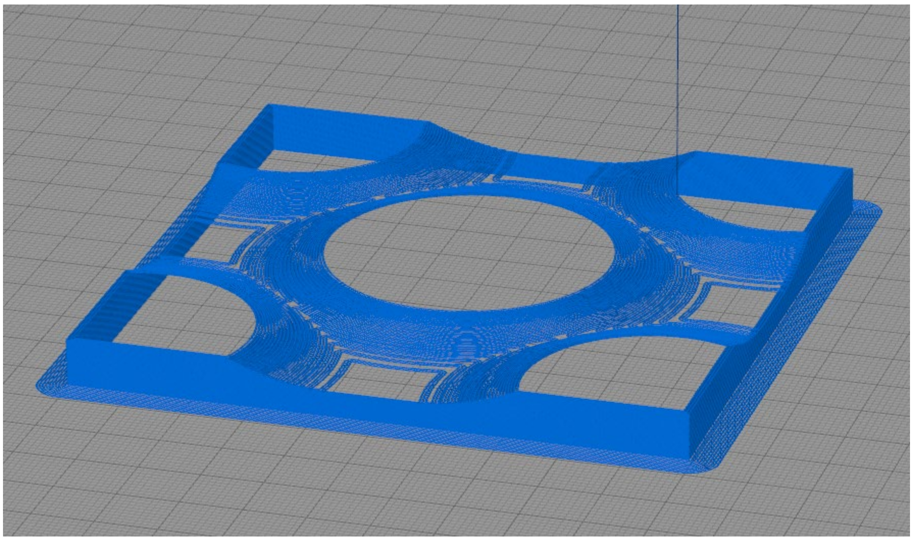

Figure 6.3.2. Iteration 2 where the generation of the planar massing layers was created, viewed in Simplify3D.

In this instance, Droid recognized that the part was dipping bellow the virtual build surface it had created and assumed it didn’t have a bottom layer. With no bottom layer, Droid was not able to provide the infill, as it had been directly linked to whether the part was considered open (having holes/gaps in the mesh) or closed.

6.3.3. Iteration 3 – Separate files

In this iteration of the non-planar workflow, came a weird and bug that arose for seemingly no apparent reason. While attempting to save the final file with both the non-planar and planar layers together, a separation issue occurred. Each set of lines that made up the sliced geometry had been broken up into their individual segments and saved in their own STL file.

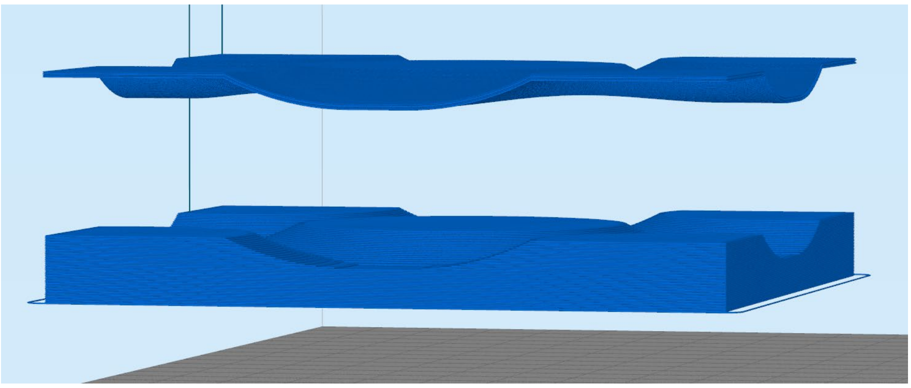

Figure 6.3.3. Separated files of planar (bottom) and non-planar (top) layers placed into one G-code viewer.

This was quite a confusing issue, at first, as all the line segments had to be fed into a component that would compress them into one linear script that would output the final G-Code for the printer to understand. While this was what was expressed in the component's description, this was not the case. A quick look at the data being output showed a clear separation of information that was being saved. A simple condensing, using the flatten function (Combines all data branches into a single branch/list), of the data being sent out was enough to fix this issue.

6.3.4. Iteration 4 – Collisions and Final Design

Up until this point, my focus was attempting to create a system that would create coherent non-planar lines that could be mapped onto any shape. While this was very successful, I neglected a key idea that would make or break this printing strategy, collisions. The g-code generation at this point needed a collision detection system that would prevent the print head from colliding with the part being printed.

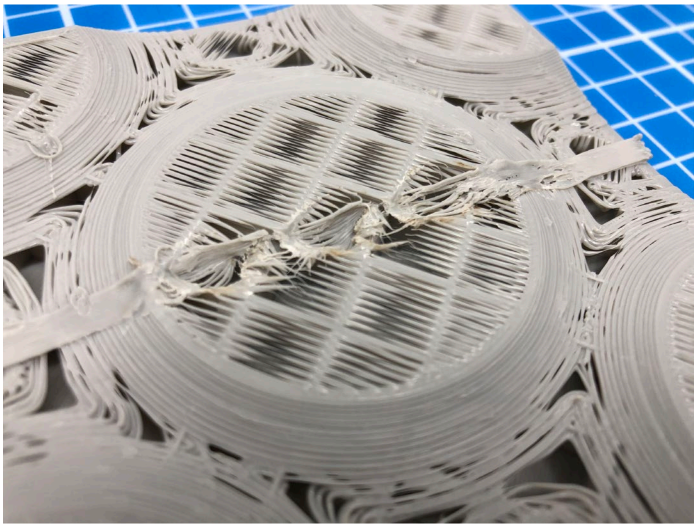

Figure 6.3.4.1. The non-planar layers (centre) are applied in the appropriate way but collide with the model during the travel paths.

Luckily, by using contours, the angle of each point on the line could be used to check whether a collision would occur, and subsequently generate an avoidance tool path movement. Preventing impacts with the already printed layers. Thus, with this new G-code now generated, it is possible to print non-planar, layers onto any simple 3D model.

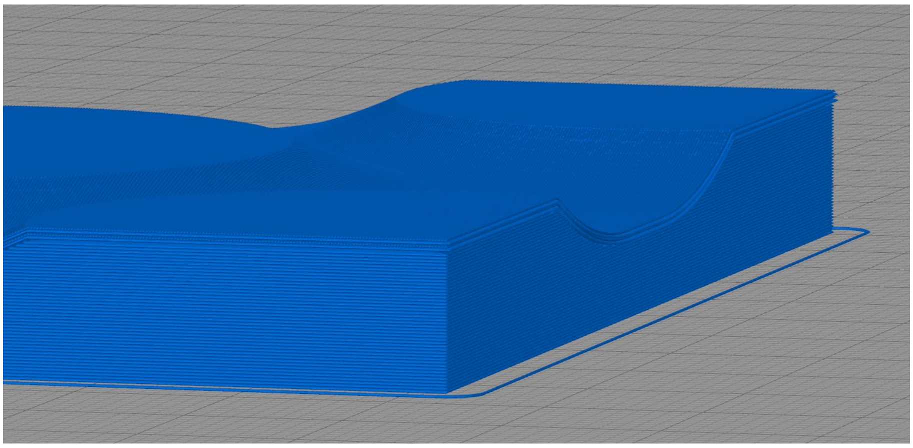

Figure 6.3.4.2. Final non-planar surface applied to the traditional planar massing, to create a hybridised system.

7. Discussion (evaluation and significance)

Throughout this research task, the key focus had been the development of a more accessible non-planar workflow for individuals in the AEC industry who had little to no complex computational understanding. As such, a consistent documentation of the iterative design process was necessary. By understanding what made the initial design too complex for users, a steady development of a more user-friendly workflow became apparent. This strategy, which once took hours of time and effort to achieve the settings for a limited few, has become much more comprehensible to a more mainstream audience.

This research papers goals where to develop a workflow that could be easily accessible by any individual with little to no computational knowledge and provide a faster alternative to produce faster and smoother surface finish on 3D models. As such, I believe that I have successfully been able to provide a working basis for an easier alternative to non-planar workflow generation, requiring very little technical understanding. By hosting the work in progress script to an open source plugin website, I have successfully made it far more accessible to a broader audience than previous methods.

However, due to time constraints and a faulty 3D printer that had to be repaired, I was unable to further my research into refining the non-planar process. As such, only objects with a flat base can be printed. Despite this, within the allotted 11 weeks of research and development, I can say that I have comprehensively been able to create, a more accessible non-planar workflow than what was possible by other researchers who had a longer research time, of approximately 2 years. This research project presents a clear and highly customizable foundation for future research into this topic. Providing anybody the opportunity to further solve many of the questions still left to answer about Non-planar printing.

One issue that I was unable to further my research on was, combining this system with support structures, and sequencing the printing process to avoid collision. An issue that was originally noted by Daniel Ahlers is his work in non-planar line generation. On top of this a more efficient collision detection method should also be further researched to prevent the nozzle from colliding with other parts on the print bed. One final issue that I have still yet to resolve comes in the form of sequencing of the layers. This is in reference to the process by which the steeper portions of the print are replaced with planar layers and, as the angle lessens, non-planar layers can resume. With more time and resources, and my script as the basis, these issues can be resolved. Understanding the problem of sequencing the layers for optimal printing, as well as a more efficient collision detection system, are the key issues that will require further testing in future research projects.

8. Conclusion

Developing a more accessible computational workflow for Non-planar printing over current g-code generation techniques can produce smooth, stronger parts using fewer resources, at a lower cost, and in a more efficient time frame. The generation of these non-planar layers works as intended. The script that was created throughout this work is stable and usable. Providing smoother upper layers to complex and freeform surfaces. Through the action research methodology, this research has explored a more efficient means of creating a Non-planar workflow that is more accessible to a broader range of individuals. The iterative design of this project provided, a means of testing key ideas, and rectifying many of the travel path issues that were not evident in Rhino/Grasshopper's virtual space. The creation of a user-friendly workflow allowed for clear and easy to achieve milestones to be set, when generating the non-planar script. Thus, it become increasingly apparent that a non-planar workflow could be achieved and made simple for other users. With this script as its basis, another research project can be conducted into non-planar G-code generation, making the process more efficient and less time consuming to achieve for the average user. In this way, better looking, and more structurally stable parts can be created. On complex parts, however, the non-planar surfaces are often not printable, due to possible collision with the print head. While some aspects may not work as intended, the script should be used with some caution as is currently considered a work in progress. In future designs, a unique nozzle and printed head could be designed to specifically work for creating non-planar layers, reaching steeper angles that would otherwise be impossible with current systems. On top of this, the bonding between the layers should increase when non-planar layers are applied above them, however, due to time constraints, this was not tested. This work is completely open source and can be found in Food4Rhino under the title **Griffin**, with the following URL: **https://www.food4rhino.com/resource/griffin**

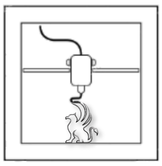

Figure 8. Griffin logo currently available on Food4Rhino

Acknowledgements

Thank you to Nicole Gardner for actually providing assistance with much of the concepts, Hank and Yannis for providing me with interesting pathways to develop, Cristina and Narridh for the constant moral support, Alec the "Grasshopper Wizard" for helping solve the countless issues that existed in the script, Daniel Ahlers for his extensive research into Non-planar printing and for providing the foundations for which this research was conducted, the CODE cohort whose constant support helped me through some tough spots in my research. And finally sleep for always being so elusive when I needed you the most.

References

3D Hubs. (2019). Introduction to FDM 3D printing | 3D Hubs. [online] Available at: https://www.3dhubs.com/knowledge-base/introduction-fdm-3d-printing/

3D Hubs. (2019). Introduction to SLS 3D Printing | 3D Hubs. [online] Available at: https://www.3dhubs.com/knowledge-base/introduction-sls-3d-printing/

Ahlers, D. (2019). 3D Printing of Nonplanar Layers for Smooth Surface Generation. [online] Tams.informatik.uni-hamburg.de. Available at: https://tams.informatik.uni-hamburg.de/publications/2018/MSc_Daniel_Ahlers.pdf

All3DP. (2019). Stereolithography (SLA 3D printing) – Simply Explained | All3DP. [online] Available at: https://all3dp.com/2/stereolithography-3d-printing-simply-explained/

Burns, M. (1993). Automated fabrication: improving productivity in manufacturing. Prentice Hall.

Chakraborty, D., Aneesh Reddy, B., and Roy Choudhury, A. (2008). Extruder path generation for curved layer fused deposition modelling. Computer Aided Design, 40(2):235–243.

Huang, B. and Singamneni, S. (2014). A mixedlayer approach combining both flat and curved layer slicing for fused deposition modelling. Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture, 229(12):2238–2249.

Instructables.com. (2019). [online] Available at: https://www.instructables.com/id/How-to-Build-Your-Own-3D-Printing-Slicer-From-Scra/

Khurana, J. B., Dinda, S., and Simpson, T. W. (2017). Active-z printing: A new approach to increasing 3D printed part strength. Solid Freeform Fabrication Symposium.

Kubalak, J. R., Wicks, A. L., and Williams, C. B. (2018). Using multi-axis material extrusion to improve mechanical properties through surface reinforcement. Virtual and Physical Prototyping, 13(1):32–38

Llewellyn-Jones, T., Allen, R., and Trask, R. (2016). Curved layer fused filament fabrication using automated toolpath generation. 3D Printing and Additive Manufacturing, 3(4):236–243.