

## FULLY THREE-DIMENSIONAL TOOLPATH GENERATION FOR POINT-BASED ADDITIVE MANUFACTURING SYSTEMS

Maxwell K. Micali\*; David Dornfeld\*

\*Department of Mechanical Engineering; University of California, Berkeley; Berkeley, CA  
94720

### Abstract

While additive manufacturing and 3D printing achieved notoriety for their abilities to manufacture complex three-dimensional parts, the state of the art is not truly three-dimensional. Rather, the process plans for the majority of these machines rely on a stack of discretized, two-dimensional layers, which results in parts with stair-stepped surfaces, as opposed to being smooth and freeform. This work proposes a change to the 2.5D paradigm by using a configuration space approach to enable toolpath planning in a full three-dimensional space, allowing movements beyond planar slices. Algorithms are also presented to compute toolpaths on non-planar regions. Since the toolpaths take part and machine geometries into account, they are guaranteed to be collision-free. These techniques are relevant to many additive manufacturing technologies, including fused deposition modeling (FDM), directed energy deposition (DED), material jetting, and nozzle-based variants. The result of implementing non-planar toolpaths is smoother, more continuous part surfaces.

**Keywords:** CAM, additive, process planning, accessibility, configuration space

### Introduction

Additive Manufacturing (AM), otherwise known as solid freeform fabrication or 3D printing, has seen a large amount of growth since its inception a few decades ago, both in terms of research activity and industrial adoption. According to the Wohlers Report, the AM industry was already valued at \$3 billion in 2012, and is projected to grow to over \$6.5 billion by 2019 [1].

There are a variety of AM processes in use, with many more names than there are processes. In an attempt to standardize the nomenclature of AM processes, the ASTM International Committee F42 on Additive Manufacturing Technologies identified the seven major types of AM processes in its "Standard Terminology for Additive Manufacturing Technologies" in 2009, which are defined as follows [2]:

- *Binder jetting* – an additive manufacturing process in which a liquid bonding agent is selectively deposited to join powder materials
- *Directed energy deposition* – an additive manufacturing process in which focused thermal energy is used to fuse materials by melting as they are being deposited
- *Material extrusion* – an additive manufacturing process in which material is selectively dispensed through a nozzle or orifice
- *Material jetting* – an additive manufacturing process in which droplets of build material are selectively deposited

- *Powder bed fusion* – an additive manufacturing process in which thermal energy selectively fuses regions of a powder bed
- *Sheet lamination* – an additive manufacturing process in which sheets of material are bonded to form an object
- *Vat photopolymerization* – an additive manufacturing process in which liquid photopolymer in a vat is selectively cured by light-activated polymerization

A selection of alternative names to these processes is shown in [2] in order to provide a mapping of some popular process names onto the F42 nomenclature. The term “rapid prototyping” has historically been one of the most popular terms for AM in general when prototyping was the primary use of the technology. While prototyping still remains an important application, AM technology has progressed to the point where direct part production is possible and increasingly common in a wide selection of build materials.

Given the fact that all of these seven core processes merit separate categorization, there are fundamental differences with each process. These differences tend to manifest in the underlying physical mechanisms driving the process, the class of materials available for the process, and the form of the materials prior to the process. However, despite these important distinctions, the digital pipeline from computer-aided design (CAD) to print is identical for all seven AM process technologies. The generalized pipeline begins with a digital CAD model, then a slicing algorithm is used to slice the CAD model into a finite number of planar slices along some axis. Information about the slices is sent to a more specialized algorithm which plans how a specific printer will print each slice, then the printer sequentially prints all of the slices into a cohesive structure. It is important to note that the resulting part represents an approximation to the original CAD model as opposed to the exact geometry, since the part is made with a finite number of slices which are in effect sampling the solid model. More on the topic of planar slicing and its associated issues will be discussed later in this paper.

With such variety in AM process technology, it is worth challenging the trope that this generic digital pipeline is optimal for all processes. The intention of this work is to investigate the merits of process-specific pipeline specialization, in hopes that specialization will enable the unique capabilities and benefits of each process to be leveraged in ways which could not be accomplished with a generic solution. More specifically, this work proposes an alteration to the digital pipeline for directed energy deposition, material extrusion, and material jetting additive manufacturing systems, which will henceforth be referred to as “point-based” additive manufacturing systems. These are point-based technologies because they all have pointed nozzles which eject material; they deposit material selectively and exclusively at the nozzle aperture; and they allow for voids within the total build volume.

### **Conventional Approach**

As mentioned previously, the digital pipeline preceding additive manufacturing processes is typically identical, regardless of the type of manufacturing process being used. The pipeline starts with a 3D solid model, which is then discretized into a finite number of flat slices with the same axial orientation, effectively reducing the sliced representation of the geometry to an approximation of the solid model. These slices are then printed sequentially by the machine,

yielding a physical part which is also an approximation of the intended solid model. If differentiation does occur in the digital pipeline for a certain process technology, it is merely in determining how the particular machine at hand can most optimally produce each slice, typically optimized for throughput or process stability.

There is a compelling reason for why the solid model processing pipeline for AM systems has little variation, and as such why computer-aided manufacturing (CAM) packages for AM offer little more than slicing algorithms. This reason is rooted in the fact that additive manufacturing systems are capable of producing highly complex and intricate geometries, and slicing algorithms are more computationally efficient with handling complexity than featurizing the part's surfaces and interior, which could also be complex internal lattices. In addition, slicing algorithms require little supervision from the user, while more sophisticated CAM approaches such as those for conventional computer numerical control (CNC) machining require an experienced user to make a series of decisions throughout the CAM stage of the pipeline.

Another advantage of flat slices comes into play when considering the motion of the machines during the printing process. Most formats of AM systems have a mechanical element traversing over the top surface of the work in progress. By only processing material in flat slices, it guarantees that the traversing machine element will not collide with the work underneath, since any given slice has a known maximum height. Changes in the machine element's height are monotonically increasing, and it is always higher than the highest slice at that stage of the process. Collision-free movement is important for any CNC process, since any collision has the potential to damage the machine and/or the part.

Conveniences of slicing aside, it also comes with its set of drawbacks. The first drawback is that the height of the machine element is not changed during any given slice, which means AM machines are only 2.5D processes. The 2.5D nature of the process results in quantifiable error when compared to the intended solid model. Error can be minimized with thinner slices, however there is a finite limit to how thin a slice can be. This limit is based on machine resolution or unrealistic manufacturing times, and since this limit always exists, slicing error will always exist whenever there are curved surfaces along the slicing axis. The error appears as the familiar stair-stepped pattern on the surface of additively manufactured parts, which is often an undesirable aesthetic quality, and results from either under filling or overfilling material with respect to the design surface.

Further complicating the use of flat slicing algorithms is the entry of new, 5-axis point-based AM machines in the market. In addition to movement in the  $x$ -,  $y$ -, and  $z$ -axes, they allow for the rotation of the build platform relative to the  $z$ -axis, as well as the tilting of the nozzle axis relative to the build platform. In a system possessing these additional degrees of freedom, the question about how to slice the solid model in order to take advantage of the system's extra capabilities becomes nontrivial, and if that question is not asked, the additional degrees of freedom add little value to the process. This question can also be interpreted as an opportunity to move beyond flat slices altogether and implement something more sophisticated. Figure 1 shows a summary of the current process and the need to implement new solutions. Figure 2 uses a diagram to demonstrate two types of error resulting from slicing algorithms representing a solid model with a curved surface.

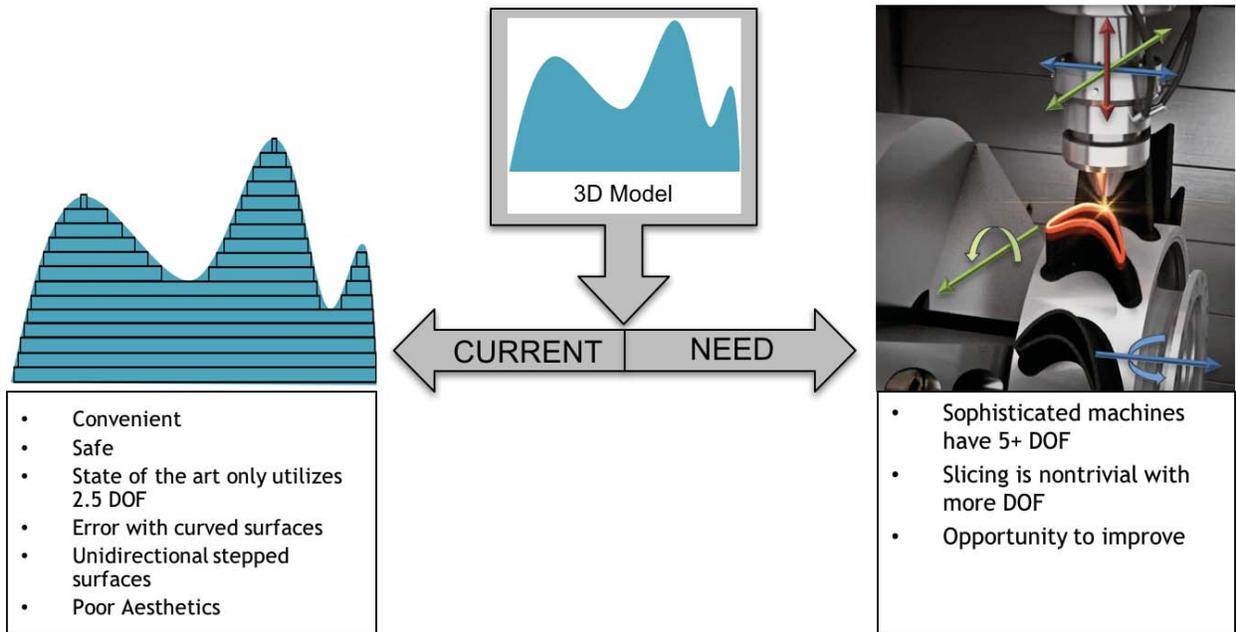


Figure 1: This figure highlights the crossroads faced by AM, where the current, flat slicing processes have fundamental limitations, especially when considering the requirements for systems with more than three axes of movement.

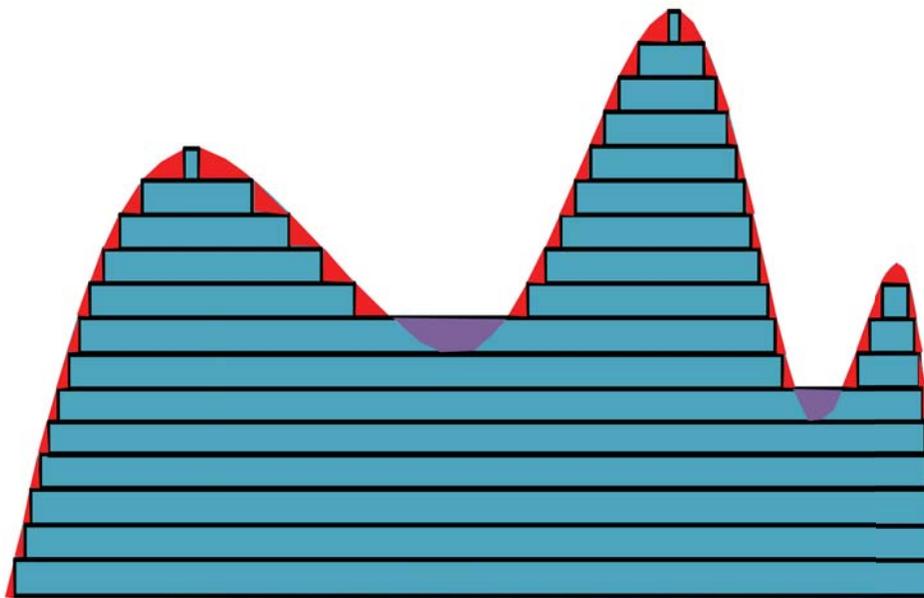
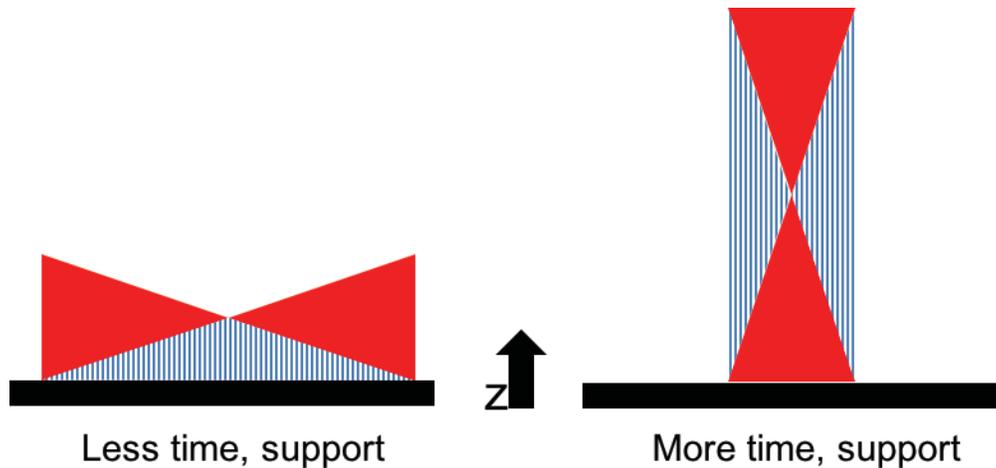


Figure 2: A diagram representing two types of error which can result from slicing a solid model with curved surfaces. The regions shaded in red show where the slice is locally undersized compared to the design surface, and the regions shaded in purple show where the slice is oversized compared to the design surface. The red errors result in 'gouges', while the purple errors result in depositing material in regions beyond the design surface.

There have been widespread efforts to optimize particular qualities of additively manufactured parts and the build processes through algorithmically selecting a part's most ideal orientation on the build plate prior to slicing, as depicted in Figure 3. The following is an incomplete list of traits which are functions of a part's orientation:

- the total time required to build the part
- the amount of support material required and its contact area with the part
- the contact area between the build plate and the part
- the mechanical properties of the part resulting from the layering direction
- the total amount of error resulting from slicing



*Figure 3: Part orientation on the build plate can impact qualities of both the part and the process. Shown here is a simplified 2D projection of a part (red) being built in two different orientations. Since increases in height along the Z axis take longer to print, the orientation on the left takes less time to print even though the part geometry is identical. In addition, the part on the right requires more support material (blue) in order to build the overhanging structure.*

It is common currently for slicing programs to include a function to calculate the best part orientation to optimize one or several of the first three bullets listed above. While there have been some research efforts to select the orientation in order to minimize total volumetric error of the slicing direction, these approaches are still affected by issues inherent to parts composed of flat slices, such as highly anisotropic mechanical properties [3].

Discretizing a solid model into nonplanar slices rather than flat slices could improve the total volumetric error, as well as allow more control over the anisotropy of the resulting structure. In order to test these hypotheses, there must first be a way to program an AM machine to follow nonplanar slices without collision. The research detailed in this work addresses the question of how to move toward fully three-dimensional toolpaths for point-based additive manufacturing systems and away from the current 2.5D flat slicing approach to process planning.

### **Relevant Literature**

This research began by taking a step back and viewing additive manufacturing processes under a more general lens, as opposed to associating them with slicers by default. It became

obvious that point-based AM systems have many features in common with CNC milling machines. The most salient similarities are how the machine is given instructions, how the machine moves, and how well-defined and rigid the tool is inside the machine. Following, strategies used for toolpath generation in the field of CNC machining can be relevant literature for fully three-dimensional additive manufacturing toolpaths. For the remainder of this paper, it is assumed that “additive manufacturing” is referring to point-based additive manufacturing systems unless otherwise specified.

Another similarity between AM and CNC milling can be seen when comparing the surfaces of a printed part and a part which has been through roughing cuts on the mill. If the design geometry is curved, both part surfaces will have a stair-stepped pattern. In this case the milling machine is also operating on geometry which has been discretized into slices – however, the surface does not remain this way in a completed milled part. Roughing is just a precursor to the finishing passes for machining complex, freeform surfaces. While the roughing operations may operate in 2.5D, the finishing passes trace the contours of smooth surfaces, requiring fully three-dimensional toolpaths. This process of a milling cutter tracing intricate, smooth surfaces is called Sculptured Surface Machining (SSM), and it is a technique commonly used in machining die-cavities. Methods used in generating finishing toolpaths for SSM are important to consider in generating toolpaths for nonplanar AM.

SSM is a point milling process where the milling cutter moves through a series of cutter contact (CC) points, which are points where the cutter is in contact with the design surface. Another way to parameterize the positions of the cutter is to use cutter location (CL) points. CL-points are unique for each position of the cutter, and are defined based on some reference point on the tool, such as the spherical center of a ball nose end mill. When the tool is slid over the design surface of the part, the CL-points form a CL-surface. A diagram of the CL-surface is shown in Figure 4. Since the tool cannot be slid over an unmachined stock surface to form the CL-surface for generating a design surface, other methods must be used to calculate the CL-points a priori.

The CL-points can be calculated locally by using a CC-point approach or a direct positioning approach. The CC-point approach first plans the toolpaths on the design surface, then computes the CL-points from that set of CC-points. Sata et al. [4] used an isoparametric method to plan the CC-points, but difficulties arose with the non-uniform relationship between the parametric coordinates and the physical coordinates [5]. Huang et al. [6] proposed an algorithm based on nonconstant parameter tool contact curves, and Suresh and Yang [7] proposed isocurvature methods along characteristic lines. A direct positioning approach can also be used, which plans 2D toolpaths first on the horizontal guideplane, then lowers individual tool positions until they touch the design surface [8].

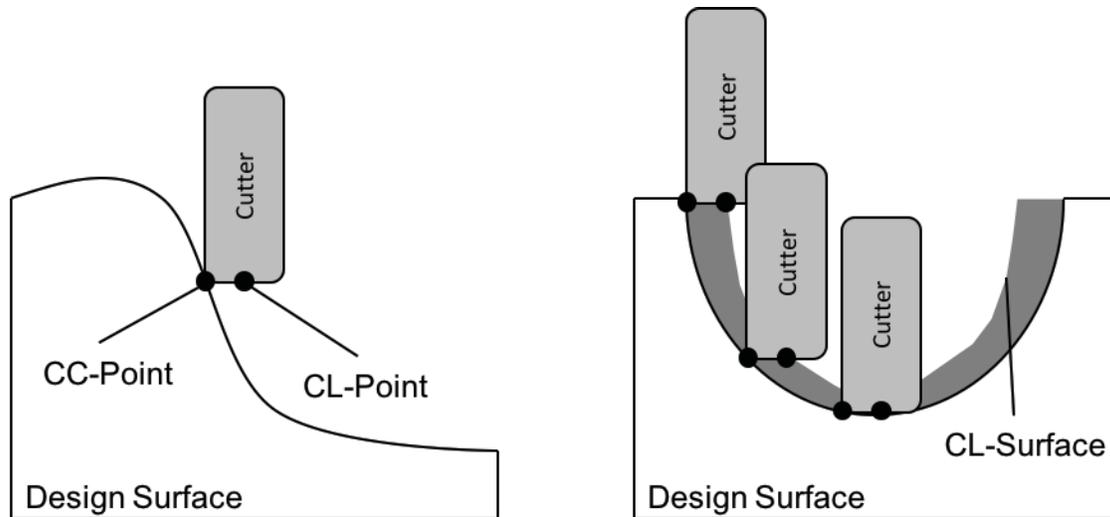


Figure 4: Diagram showing the construction of a CL-point and a CL-surface

Local toolpath generation methods are not ideal because they are prone to gouging. Two locally safe CL-points could be linked to form a CL-line which is not a gouge-free path over the design surface, and this is difficult to anticipate using local methods. In the context of high speed machining, any gouge or collision could be disastrous. As a result, the needs for high speed machining drove Choi et al. [8] to develop a configuration space (C-space) approach to toolpath generation, which is a global approach and therefore less prone to gouging and collisions than local methods.

C-spaces are a concept used in robotic spatial planning problems, especially for automatic manipulator movements [9]. Each element of the C-space corresponds to a valid system configuration, and these configurations can be described by whether or not they are safe. Safety is defined as the set for object  $A$  and obstacle  $B$  where  $A \cap B = \emptyset$ . Applying C-spaces to cutting tools and the problem of toolpath generation yields some convenient properties, since it guarantees gouge-free and collision-free paths. In order to accomplish this, the first step in the toolpath generation process is to transform the design surface into C-space elements. Once this is done, the toolpaths are generated within the C-space elements rather than on the design surface, which automatically incorporates information from the global part geometry in the path planning process.

A convenient C-space transformation to use is the inverse tool offset (ITO), shown in Figure 5. The ITO is constructed with the Minkowski sum concept. This works out in practice by inverting the tool, locating the CL-point of the inverted tool on the design surface, then sweeping out an envelope of the inverted tool's geometry while moving along the design surface. The envelope is the ITO-surface, and it also defines a safe C-space for the tool. During operation, the ITO-surface is the CL-surface required in order to machine the design surface. Thus, toolpaths can be generated within the ITO-surface to have effective, yet gouge-free and collision-free operation.

It is worth noting some key differences about AM which must be taken into account if a similar ITO method is to be used in a C-space approach to toolpath generation. With CNC

machining, it is assumed that if a cutter contacts a surface, the cutter will machine the surface, provided the CC-point(s) are on a cutting surface of the tool for a given CL-point. In addition, contact with a cutting surface on the tool is sufficient to consider machining to have occurred. In AM on the other hand, which will be assumed to have nozzles which can be modeled with a right cone, only a very specific point of the nozzle can deposit material – the tip. Thus if the tip is not in contact with the design surface but the side of the nozzle is in contact as a result of the ITO, this is not contributing anything to the deposition process. Rather, it is a sign that the tool has approached its limitation in that region of the part. Also it is not enough to simply program the position of an AM tool, since the material deposition process must be programmed as well.

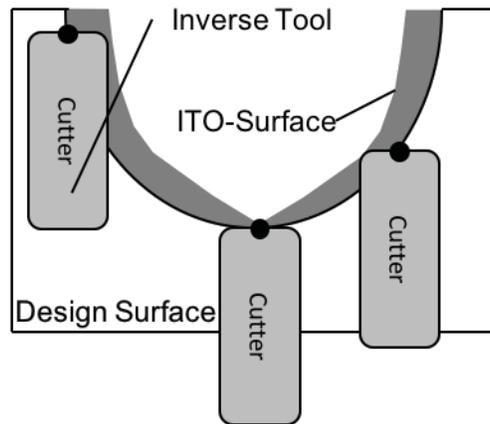


Figure 5: Inverse tool offset (ITO) surface

### **XY-Parallel Toolpath Generation for Three-Axis, Point-Based Additive Manufacturing Systems**

Choi et al. in [8] proposed an XY-parallel one-way finishing unit machining operation (UMO) algorithm as a proof of concept for taking a C-space approach to SSM. A logical extension of that algorithm with appropriate enhancements and modifications can serve as a proof of concept for applying these approaches to AM toolpath generation. That proof of concept is the subject of this work. Figure 6 shows a schematic of the overall toolpath generation algorithm for XY-parallel toolpaths for three-axis AM systems. It begins with the user specifying a minimum number of parameters: the design surface geometry, the geometry of the nozzle, the tolerance of the process or the minimum resolution of the machine, and the maximum step-length allowed along a path. The algorithm proceeds first by generating an ITO-surface from the design surface and the nozzle geometry. Since the ability of conical nozzles to access narrow regions decreases with the aspect ratio of the nozzles' height to its base diameter, the printability of the part is a function of whether a given nozzle can access all of a given design surface. Unless the entirety of the design surface is accessible, the part cannot be printed as intended. To address this issue more concretely, a printability check is incorporated in the algorithm after the ITO-surface is generated. If the part is deemed unprintable, the algorithm will inform the user that the aspect ratio of the nozzle is too low for the given design surface. Otherwise, the process will proceed to the next step of defining XY-parallel paths within the C-space, followed by calculating control commands for the material deposition function of the machine. All of this information is finally synthesized by a script that converts it into a standardized G-code format

which AM systems can understand, provided the proper machine-specific heading is used at the top of the file. Since few AM machines support circular interpolation commands (G02 & G03), the types of output commands for this toolpath generation algorithm are limited to linear interpolation (G01) and rapid traverse (G00).

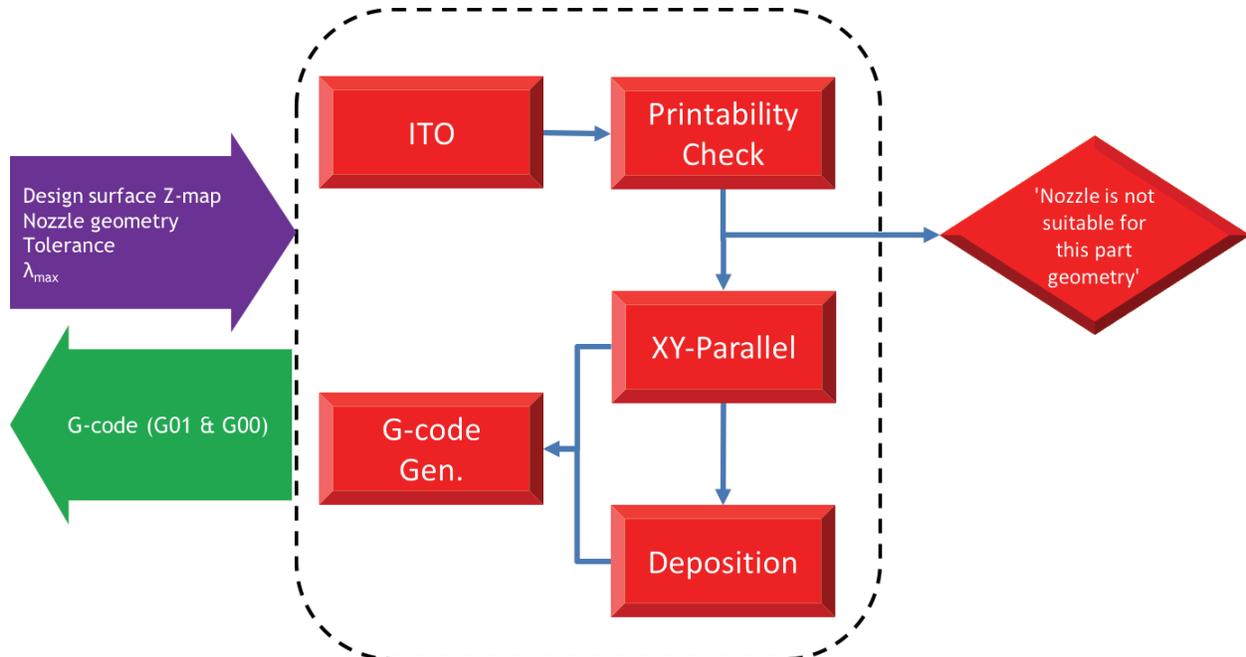
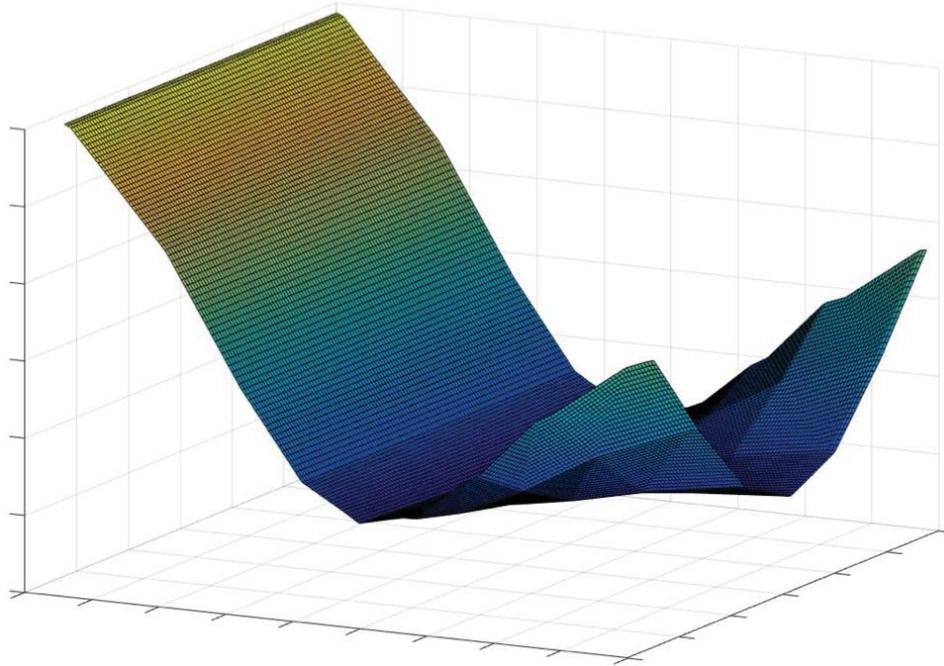


Figure 6: Schematic of XY-parallel toolpath generation algorithm for three-axis AM systems

The design surface used to test these algorithms is shown in Figure 7 [10]. The format of this surface is a Z-map, which is a matrix of height values in an equally spaced grid corresponding to the X and Y indices of the matrix. The doubly curved nature of this surface makes it particularly apparent whether the toolpaths can trace the surface curves.



*Figure 7: Design surface used to test algorithms*

Generating the ITO for a nozzle modeled as a right cone proceeds similarly to how it does for other tools seen in CNC machining. It starts by inverting the cone, then placing the nozzle location (NL) point along the design surface, sweeping out an envelope of the nozzle body during the process. A diagram of this can be seen in Figure 8 below. Terminology is changed to NL-point from CL-point to be more consistent with the process, since this nozzle does not cut any material. The red cones are inverted nozzles, and the purple shaded regions depict the portion of the ITO-envelope which exists above the design surface. The nozzle in this figure has too low of an aspect ratio to access this entire surface, as shown by the blue triangles in the figure which are at the edge of the allowable envelope but do not have their tips in contact with the design surface. In order to reach the design surface with the nozzle tip in these regions, the nozzle would need to intersect with the surface, and this is not an allowable configuration in the safe C-space. Since the sides of the cone do not deposit material, this nozzle would not be able to print this design surface.

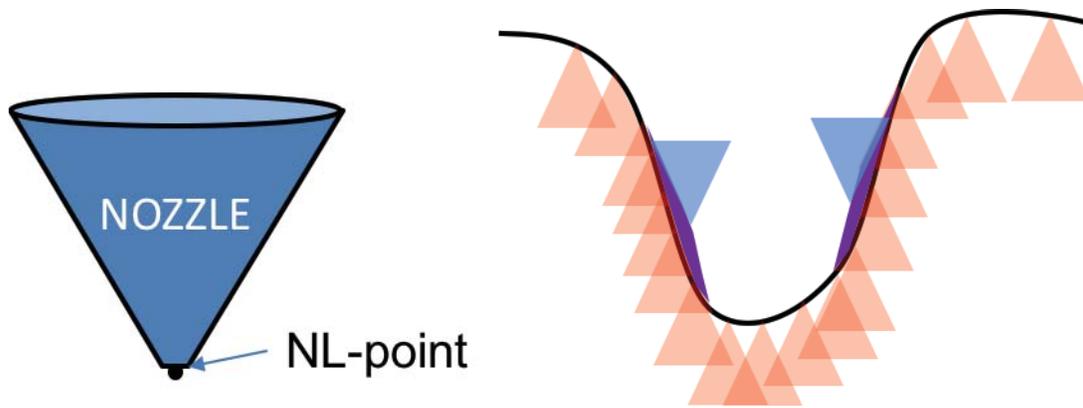


Figure 8: (left) Diagram of nozzle and its nozzle location (NL) point. (right) Diagram of the ITO-surface generation process with a right cone tool geometry on an arbitrary design surface. This cone's aspect ratio is too low to fully access this design surface. The purple regions indicate the zones inaccessible by the nozzle tip.

It is possible to calculate whether a given nozzle geometry would be capable of printing the entirety of a given design surface, and one approach to calculating this is incorporated within the present algorithm. Since the NL-point corresponds to the tip of the nozzle and this point must be able to access any point of a design surface for it to be fully printable, any separation between the ITO-surface and the design surface which is greater than the manufacturing tolerance at any given point can render that nozzle/surface combination “unprintable”. The ITO in this application has the most value for being a convenient method for algorithmically determining printability. See Figure 9 for a visual depiction of the NL-surface and design surface delamination due to a nozzle aspect ratio error. A field showing the local magnitude of separation between the two surfaces is plotted beneath the surfaces. From the magnitude field and the NL-surface, it can be deduced that there is a maximum allowable slope for the movement of the nozzle within the safe C-space, which is consistent with what would be expected for a vertical cone traversing a surface. The solution to a nozzle aspect ratio error like the one shown is to use a nozzle with a higher aspect ratio in order to enable the manufacturing of steeper surfaces. Even with an infinite aspect ratio nozzle, three-axis AM systems will not be capable of producing a vertical feature within a single, curved layer.

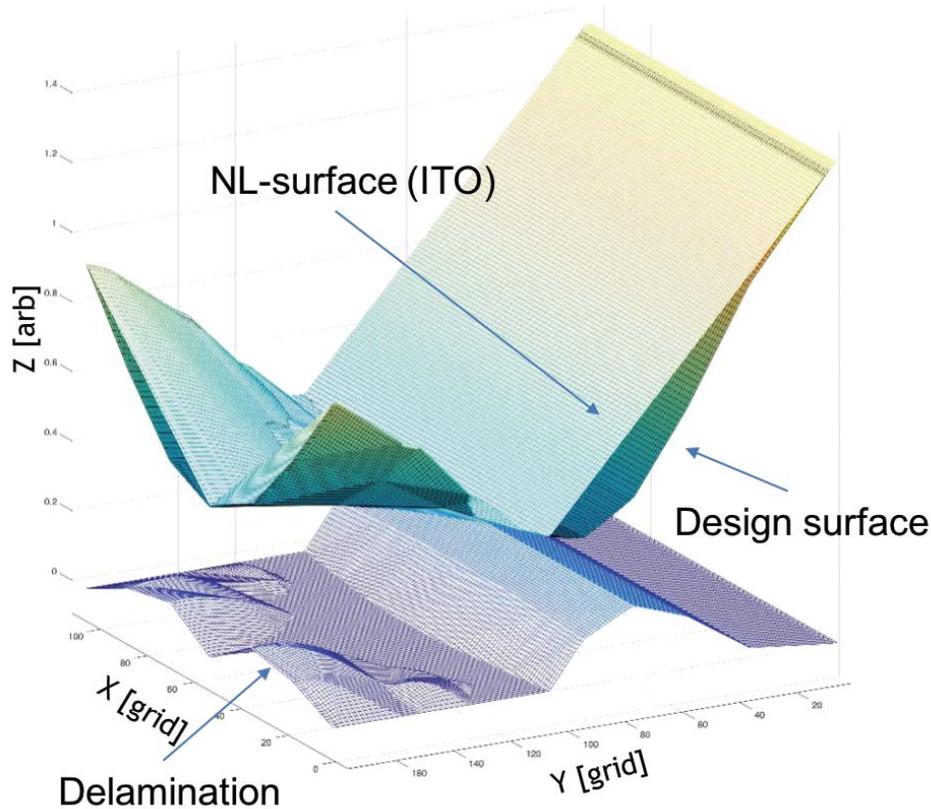


Figure 9: Delamination of the NL-surface and design surface due to a nozzle with insufficient aspect ratio for the given design surface. The NL-surface and design surface are plotted on top of each other, and a field showing the local magnitude of their difference is plotted below them.

Once the ITO has been generated and the printability is verified, it is time to generate XY-parallel toolpaths. Since the toolpaths are calculated within the safe C-space, it is guaranteed to be collision-free. The algorithm implemented for this stage is adapted from the work done by Choi et al. for the XY-parallel paths in [8]. Pseudocode for this algorithm is below, where  $\omega$  is the path width,  $\lambda$  is the maximum step-length,  $\alpha$  is an inclination angle in either the feed or pick direction, and  $tol$  is the manufacturing tolerance. In practice,  $\omega$  is likely to be close to the aperture diameter of the nozzle, and it makes sense to set  $tol$  equivalent to the resolution of the AM machine.

```

Start at edge,  $\Delta_y = \omega$ 
  Output NL-point (X,Y,Z)
  Fit circle in feed-forward direction
   $\lambda = \min(\lambda_{max}, 2*\sqrt{2*tol*R_f})$ 
  Calculate inclination angles  $\alpha_{f,p}$ , in feed & pick directions
   $x = x + \lambda*\cos(\alpha_f)$ 
   $\Delta_y = \min(\Delta_y, \omega*\cos(\alpha_p))$ 
  Continue until opposite edge
New pass:  $y = y + \Delta_y$ ,  $x = x_o$ ,  $\Delta_y = \omega$ 
Continue until done
Linearly interpolate to form NL-lines

```

With a set of NL-lines and NL-points, it is possible to calculate commands to send with the G-code output to the part of the control system which manages the material deposition in the AM machine, such as a powder feeder or extruder motor. The following analysis will proceed as if it is an extruder motor in a material extrusion system, as shown in Figure 10, but this analysis could be reformulated to cater to a powder feeding system in a directed energy deposition system. Since the aperture at the nozzle tip is smaller than the diameter of the incoming filament, the change in linear position due to the extruder motor on the filament is amplified upon exit of the nozzle. All steps along NL-lines are linear interpolations because many AM system controllers cannot interpret higher order commands, so it is appropriate to calculate the path length between any two successive NL-points by calculating their three-dimensional Euclidean distance. Once the nozzle's path length is calculated, the required motion of the extruder motor can be calculated to satisfy that extruded path length, and the current position of the motor is updated. These calculations are shown below in Equations 1, 2, and 3 for a span between two arbitrary points  $i$  and  $j$ .  $E$  refers to the absolute position of the extruder motor, and  $d$  refers to the diameter of the given end of the nozzle. Please refer to Figure 10 for explanations of the variables. Extrusion does not occur when the nozzle is traversing to a new NL-line.

$$E_j = E_i + \Delta E \quad (1)$$

$$\Delta E = \Delta l \times \frac{d_n^2}{d_f^2} \quad (2)$$

$$\Delta l = \sqrt{(x_j - x_i)^2 + (y_j - y_i)^2 + (z_j - z_i)^2} \quad (3)$$

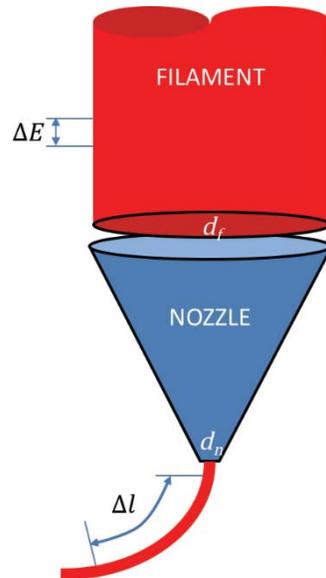


Figure 10: Schematic of a nozzle in a material extrusion system showing the material filament upon entrance and exit, as well as the relative change in length of the filament on either side of the nozzle for a given amount of extrusion.

Renderings of sample outputs for the algorithm are shown in Figures 11, 12 for both a coarse and a fine maximum step-length, respectively. The coarse step-length is ten times the maximum step-length of the fine step-length. These outputs step along the  $x$ -axis, however stepping could be performed along any direction. Near the uppermost portion of the surface, all of the parallel paths start identically. As they travel down the slope and the surface becomes more doubly curved, the paths begin differentiating and adapting to the local curvature. This is easiest to see in the coarse figure. Both surfaces look like the design surface in Figure 7, and the two step-length trials pictured produce surfaces which look similar. There must be some ideal maximum step-length for each design surface and nozzle combination, after which increasing resolution via step-length does not improve the quality of the manufactured part. This leads to the question of how to automatically determine the maximum tolerable maximum step-length, however that topic is saved for future work.

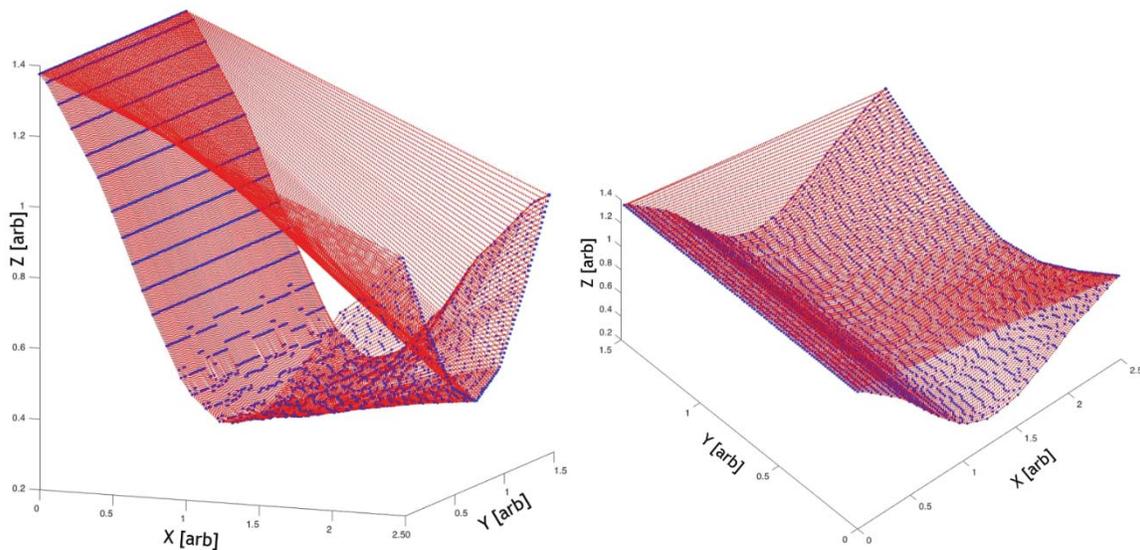


Figure 11: Two rotated renderings of sample output for the test surface with steps along the  $x$ -axis. Grid spacing of the  $Z$ -map is 0.0141 arbitrary units, and the maximum step-length is 0.1 arbitrary units. Blue dots show NL-points, and red dashed lines show interpolated NL-lines.

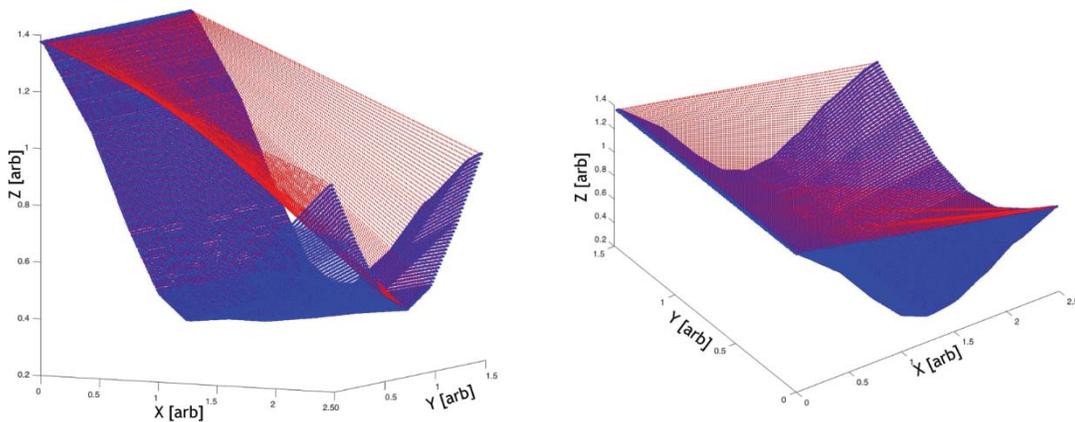


Figure 12: Two rotated renderings of sample output for the test surface with steps along the  $x$ -axis. Grid spacing of the  $Z$ -map is 0.0141 arbitrary units, and the maximum step-length is 0.01 arbitrary units. Blue dots show NL-points, and red dashed lines show interpolated NL-lines.

To see how the extruder motor behaves over time, please refer to Figure 13. The position of the extruder is monotonic increasing, which indicates that the algorithm is performing as expected. In addition, the plot of position vs. NL-point shows variability in the extruder position update, as expected since the path length with each step between NL-points is variable. There are occasionally two NL-points which have identical extruder motor positions, and this is when the nozzle traverses from the end of one parallel path to the beginning of the next one, so there should be no extrusion during this traversal.

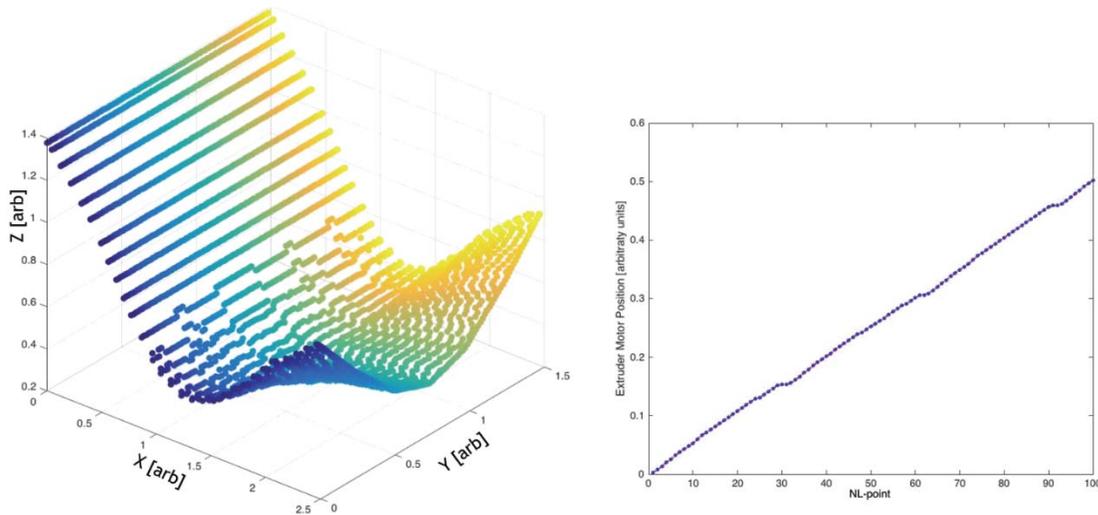


Figure 13: (left) Rendering of extruder motor absolute position shown in color at each NL-point for a  $Z$ -map with grid spacing of 0.0141 arbitrary units and maximum step-length of 0.1 arbitrary units. The position value increases from dark blue to yellow. (right) A plot of the extruder motor position for the first 100 NL-points.

### Conclusions and Future Research

The results of this work are algorithms to automatically generate of  $XY$ -parallel toolpaths for point-based additive manufacturing systems. This is one of hopefully many examples of the digital pipeline being tailored to specific additive manufacturing processes in order to leverage the unique strengths they each afford. Beyond that, these results are important because they show that a third dimension can be activated in 3D printing, and they also show that the field of additive manufacturing can draw from the toolpath generation work that has been done in the field of CNC machining with careful and thoughtful modification.

The  $C$ -space approach seems to be most fruitful as a method for determining printability for a given design surface, nozzle geometry, and manufacturing tolerance. Calculating the ITO-surface is a convenient way to determine printability, and it is important to know whether a part is printable before attempting.

By using fully three-dimensional toolpaths, the printer nozzle can trace the complex contours of a curved surface as opposed to discretizing them and approximating them with a stair-stepped appearance. This will certainly reduce the total amount of volumetric error in printed parts when compared to the intended solid model. While very steep curves are difficult even with high aspect ratio nozzles, there will be a wide set of parts that can still benefit from using this approach. This method allows a printer to effectively increase the resolution of its products without modifying the components of the printer, since this set of algorithms outputs G-code. Another benefit of converting instructions to G-code in the algorithm is that, since many AM system controllers can interpret G-code, the methods contained in this paper can benefit all of those printers which are currently in operation. While there would be more opportunities for interesting extensions to this work if machine controllers could interpret G02 and G03 circular interpolation commands, this is a hurdle which is difficult to overcome for existing machines.

Some future research projects which are in progress involve validating these algorithms on AM systems, which includes characterizing the surfaces of the final part, as well as the reduction in total volumetric error. It would be interesting to investigate how nonplanar slicing affects the mechanical properties of a part when compared to flat slicing algorithms.

The emphasis of this work is on the geometric path itself, however the machine dynamics and the underlying physical phenomena in additive manufacturing systems are also critical for successful part production. The author is looking forward to incorporating physical modeling into the toolpath generation process in future iterations and extensions of this work.

### **Acknowledgements**

This material is based upon work supported by the National Science Foundation Graduate Research Fellowship Program under Grant No. 1106400. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

- [1] “Wohlers Report 2012. Additive Manufacturing and 3D Printing State of the Industry,” Wohlers Associates, Ft. Collins, CO, 2012.
- [2] V. Pena, B. Lal, and M. Micali, “U.S. Federal Investment in the Origin and Evolution of Additive Manufacturing: A Case Study of the National Science Foundation,” *3D Print. Addit. Manuf.*, vol. 1, no. 4, pp. 185–193, Dec. 2014.
- [3] S. H. Masood and W. Rattanawong, “A Generic Part Orientation System Based on Volumetric Error in Rapid Prototyping,” *Int. J. Adv. Manuf. Technol.*, vol. 19, no. 3, pp. 209–216, Feb. 2002.
- [4] T. Sata, F. Kimura, N. Okada, and M. Hosaka, “A New Method of NC Interpolator for Machining the Sculptured Surface,” *Ann. CIRP*, vol. 30, no. 1, pp. 369–372, 1981.
- [5] P. K. Wright, D. Dornfeld, V. Sundararajan, and D. Misra, “Tool Path Planning Generation For Finish Machining of Freeform Surfaces in the Cybercut Process Planning Pipeline,” 2007.
- [6] Y. Huang and H. J. Oliver, “Non-constant parameter NC tool path generation on sculptured surfaces,” *J. Comput. Eng.*, vol. 1, pp. 411–418, 1992.

- [7] K. Suresh and D. C. H. Yang, “Constant Scallop Height Machining of Free-form Surfaces,” *J Eng Ind*, vol. 116, pp. 253–259, 1994.
- [8] B. K. Choi, D. H. Kim, and R. B. Jerard, “C-space approach to tool-path generation for die and mould machining,” *Comput.-Aided Des.*, vol. 29, no. 9, pp. 657 – 669, 1997.
- [9] T. Lozano-Perez, “Spatial planning: A configuration space approach,” *Comput. IEEE Trans. On*, vol. 100, no. 2, pp. 108–120, 1983.
- [10] S. McMains, *dsurf.txt from ME290D course*. University of California, Berkeley, 2014.