The Pennsylvania State University

The Graduate School

REDUCING NEAR-SURFACE VOIDS IN METAL (Ti-6AL-4V) POWDER BED FUSION ADDITIVE MANUFACTURING: THE EFFECT OF INTER-HATCH TRAVEL TIME.

A Thesis of

Materials Science and Engineering

by

Brett A. Diehl

Submitted in Partial Fulfillment
of the Requirements
for the Degree of

Master of Science

May 2020

The thesis of Brett A. Diehl was reviewed and approved* by the following:

Abdalla Nassar
Associate Research Professor, Applied Research Laboratory
Thesis Advisor

Jayme Keist
Assistant Research Professor, , Applied Research Laboratory

Allison Beese
Associate Professor of Materials Science and Engineering

John Mauro
Professor of Materials Science and Engineering
Chair, Intercollege Graduate Degree Program in Materials Science and Engineering

ABSTRACT

Powder bed fusion additive manufacturing (PBFAM) is being rapidly adopted by industry for the production of novel and complex geometries. However, production of PBFAM parts with near-surface voids remains a primary concern. Such voids impede the use of additive manufacturing for thin geometries and may significantly reduce fatigue life in larger AM components. Here, we develop a statistical model which relates the probability of near-surface void formation to the toolpath used in part production. Registered, computed tomography (CT) data were used as the ground truth to measure void location and morphology together with actual build plan data (e.g. vector trajectories, powers, speeds). From this, we identify the time between nonconsecutive, adjacent hatch strikes to be a statistically significant ($p = 8.54*10^{-20}$) indicator of the likelihood that a given point becomes a void. Furthermore, we show that altering the hatching strategy to include fewer short hatches (i.e. a longer time between nonconsecutive, adjacent hatch strikes) can produce a similar geometry with significantly fewer near surface voids.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

I would like to acknowledge Jared Blecher and Ryan Overdorff of 3DSystems for assisting with fabricating the parts and David Corbin for helping set up the experiment.

As well, I would like to thank my thesis advisor Dr. Abdalla Nassar for being extremely supportive and helpful during my grad school career, always quick to respond with thoughtful advice.

I would also like to thank my family for their support and encouragement throughout my years of studies.

This effort was performed through the National Center for Defense Manufacturing and Machining under the America Makes Program entitled “Generation and Characterization of Parameter- and Process-induced Defects” and is based on research sponsored by Air Force Research Laboratory under agreement number FA8650-16-2-5700. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation thereon.

Chapter 1

Introduction, Literature Review, and Experimental Background

Metal Additive Manufacturing

Additive manufacturing is, as defined by ASTM, ‘‘a process of joining materials to make objects from 3D model data, usually layer upon layer, as opposed to subtractive manufacturing methodologies.” While several metal AM processes have been developed, directed energy deposition (DED) and powder bed fusion (PBF) are the most commonly used and studied methods, both using either an electron beam or laser as an energy source. In laser-PBF, layers of powder are spread under an inert atmosphere and then selectively melted using a laser, directed by a galvanometer. Typically, on a layer-by-layer basis, the laser first traces the outline of the part (the contour) and then travels back and forth (hatches) across the interior of the component.

Laser-PBF allows for the fabrication of novel geometries, and reduces material waste and the environmental impacts of manufacturing. The ability to fabricate novel geometries of nearly arbitrary complexity also permits rapid low cost prototyping and fabrication of one-off parts. For these reasons, the technology is being adopted in the aerospace, automotive, and medical industries.

High-cost material systems, such as titanium alloys are of particular interest for aerospace and medical applications. One titanium alloy, Ti-6Al-4V, is of particular importance due to its wide range of applications and the high-cost of machining using conventional, subtractive processes. In aerospace, there is also a potential performance advantage in leveraging AM to reduce material usage and component weight. Lockheed Martin estimates that metal AM may

“save between 30 and 50% of the cost of machining aerospace titanium structural components, which are some of the most expensive components in the F-35 airframe [4].’’

AM also provides a financially-viable means for producing customizable, low-volume components. Due to the low fixed costs of metal AM (no dies or tool must be purchased to fabricate each unique part), the production of unique components or low quantity components is cheaper through metal AM than conventional methods. As a result, metal AM greatly increases the viability of patient specific Ti-6Al-4V implants, such as mesh scaffolds and porous bone-like implants.

However, AM processes face challenges such as high feedstock cost, low production rate, and size limitations. As well, components produced by additive manufacturing may exhibit reduced fatigue performance compared to wrought components, and there is a lack of standards for AM component certification.

The discrepancy in fatigue life is thought to be largely driven by micro-porosity and surface roughness, which can only be mitigated by hot isostatic pressing and surface finishing, respectively, increasing fabrication time and cost [4]. The voids nearest to the surface—which have been modeled to induce larger stress concentrations, as shown in figure 1-1—have been shown to have to greatest impact on high cycle fatigue life [4].

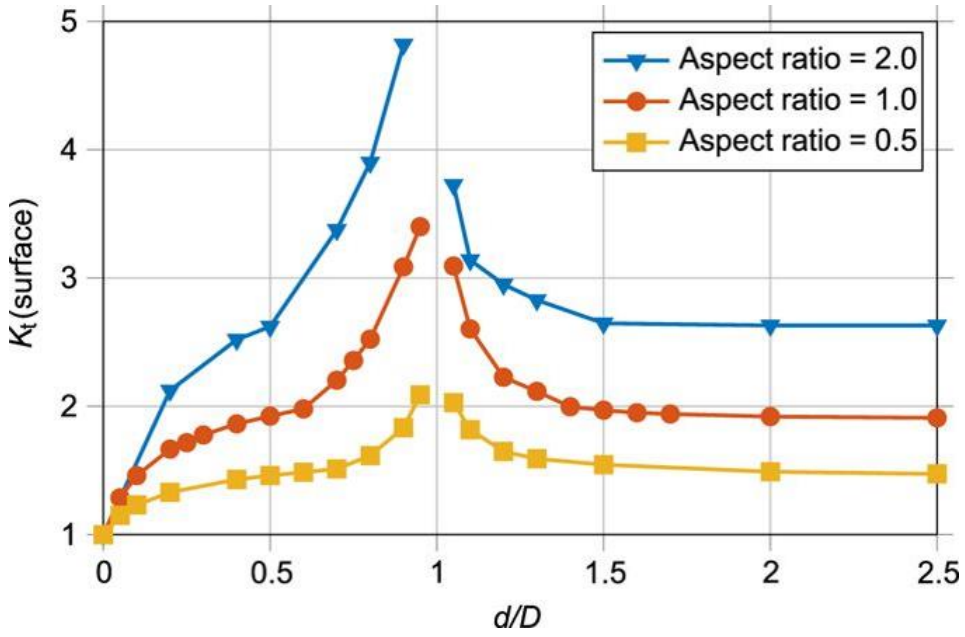

Figure **1-1**: The stress concentration (K_t) generated by a pores as a function of normalized depth. Results are shown for idealized spheres/oblate spheroids with three different aspect ratios. Adapted from [1].

As such, near surface void mitigation is essential in extending the applicability of metal AM to fatigue-limited designs.

Void Formation Mechanisms in L-BPF

Researchers have described three main void formation mechanisms: lack of fusion, keyhole collapse, and entrapment of gases in the melt pool [5]. Lack of fusion voids occur when the energy input is not sufficiently high to melt enough powder to create a solid part [5].Keyhole formation occurs when energy input is too large, causing a vapor depression which is significantly deeper than its width. The keyhole may be unstable, and leave behind small voids as its walls oscillate, or it may collapse when the energy input decreases [5]. Gas entrapment occurs

when the melt pool forms and solidifies before gases below it can escape. Because the inert gases used in L-PBF are insoluble in metals, they are unable to dissolve and pass through the melt pool.

For alloys like Ti-6Al-4V, an additional mechanism may make keyhole formation more likely. Because aluminum has a significantly lower boiling temperature than titanium [6], it may vaporize more readily from the melt pool. This vaporization exerts a downward force on the melt pool which can create a keyhole morphology on a millisecond timescale [7]. Keyholing at the end of short hatches—which may be sufficiently hotter than hatches of nominal length—may be a mechanism that creates voids in short hatches more often than in long hatches.

However, researchers have also suggested additional mechanisms to explain void formation. For example, in addition to excessive energy input [7]–[9] causing keyhole collapse [10], melt-spatter interactions have been observed to perturb the melt pool and cause voids [11]. Martin et al. have also suggested that melt pools may collapse in on themselves and produce small voids when the laser is turned off at the end of the hatch. Alternatively, if the laser is not turned off, the increase in energy density as the laser turns around will create a keyhole which collapses when the laser accelerates, leaving behind small spherical voids. [8].

Another proposed mechanism—particularly for the formation of near surface voids—involves fluid dynamics. If two adjacent melt pools are molten at the same time, it is reasonable to suspect that they will combine to reduce overall surface energy. While hatch width in the bulk of the material has been shown to remain relatively constant, it has been modeled and experimentally verified that melt pool width oscillates between adjacent hatches near the end of each hatch [12]. That is, the hatch is wider near its beginning, as it moves away from the contour, than the in-bulk hatch width—likely due to melt pools combining. The next hatch is smaller than the in-bulk hatch width when it returns to the same contour. Alternating wide and narrow hatches may lead to poorer wetting of the hatch-contour interface and leave small tear-drop shaped voids between pairs of hatches.

Because flaws may form when energy input is too high or low, researchers have sought to control the size [13] [14] and temperature [15] of melt pools in additive manufacturing by sensing the build process and modulating the inputted power and speed during the build[16]. Other strategies include measuring the temperature of the substrate prior to beginning a hatch [17], so that each hatch begins with the same thermal starting conditions.

Because shorter hatches will have less time to dissipate heat before the laser returns on the next hatch, it is reasonable to postulate that smaller hatches may cool less than larger hatches and accumulate heat over time. If this is the case, thermally isolated regions, overhangs, sharp corners, and thin geometries may require less energetic processing parameters or longer wait times between hatches to prevent voids from forming.

One factor that may affect porosity is toolpath [3]. While a typical toolpath includes a contour and then serpentine hatches across the part, researchers have experimented with various other toolpath strategies, such as “out-in” scanning—in which the contour spirals inwards continuously until the laser reaches the center of the part—or “out-in” scanning in which the laser starts in the middle and spiral outwards [2]. An illustration of an out-in toolpath is shown in figure 1-2.

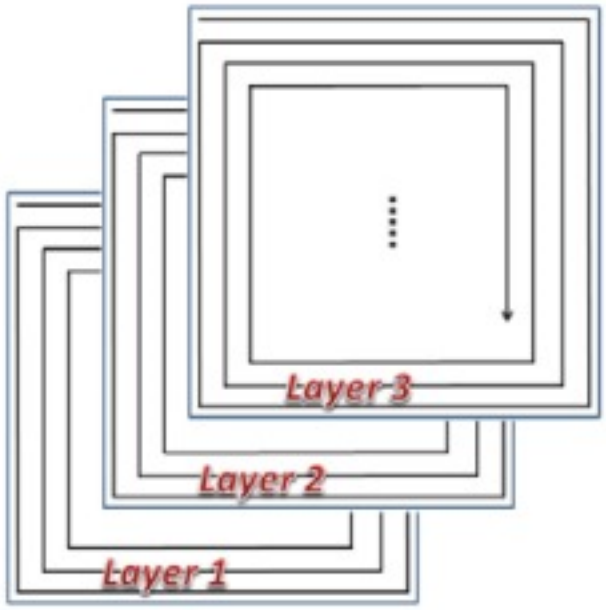

Figure **1-2**: Illustration of multiple layers of an out-in toolpath. Adapted from [2].

As well, island scanning strategies have been devised, in which the laser hatches across squares or hexagons which tessellate to fill the layer. Instead of reducing component porosity, island hatching is thought to reduce residual stresses in the fabricated component [2]. An example of an island toolpath strategy is shown in figure 1-3.

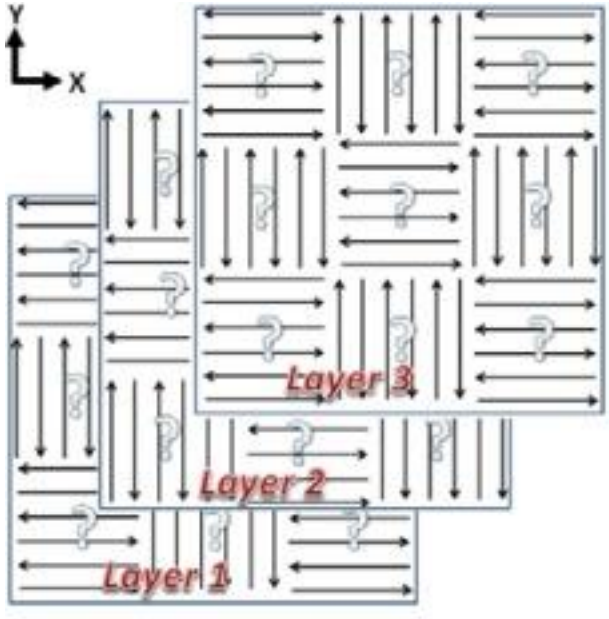

Figure **1-3**: Illustration of multiple layers of a square island toolpath. The order the squares were processed was randomly determined. Adapted from [2].

Zheng et al [18] have reported that the toolpath has a strong influence on the distribution of defects, that lack of fusion defects are most often found in between layers and tracks, and that other defects are common at the end of tracks. This is consistent with the findings of Tammas-Williams et al. that, in single directional hatching, voids form at the end of the tracks. Tammas-Williams et al. [3] have also shown that near surface voids and bulk voids exist as two distinct populations. The former is found within 150-300 microns of the contour, and the latter is found deeper than 1,600 microns within the component, as illustrated in figure 1-4.

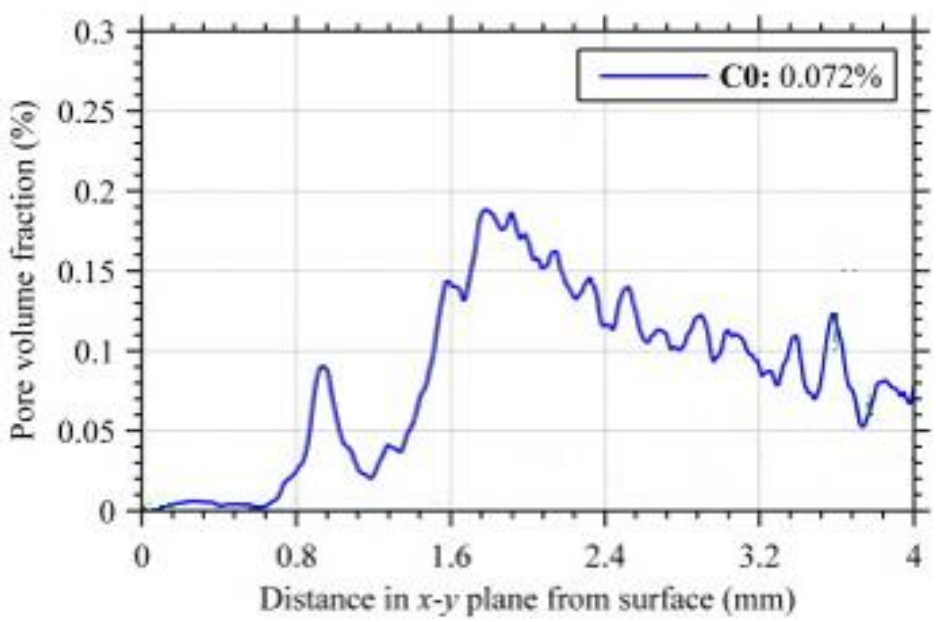

Figure **1-4**: Number of voids at a given distance from the surface, many voids are further than 1,600 microns within the component, but a 2nd population is close to the surface. Adapted from [3].

Toolpaths which reduce the occurrence of one of these two populations do not necessarily reduce the occurrence of the other, suggesting separate mechanisms for the formation of near surface voids versus bulk voids.

It has also been shown that small variations in the toolpath can results in very large changes in void number and volume, and that near surface voids are overwhelmingly more likely to form at the end of hatches than at the beginning [3]. However, a statistical prediction of near surface void formation remains largely unpursued in the literature despite the fact that near-surface voids play an important role in fatigue performance of PBFAM components [19], particularly in thin-walled geometries [20].

Motivation

There still exists a large knowledge gap on the relationship between an additively manufactured part's toolpath and the formation of near-surface voids; this is the focus of the current work. While this study does not investigate the relationship between toolpath and potential improvement of fatigue performance, it lays the groundwork by determining toolpath parameters which produce fewer near surface voids. A parameter, the inter-hatch travel time (IHTT) is identified as a strong indicator of near surface void formation likelihood. It is defined as the time needed for the laser—once at the hatch-contour interface—to move across the component and return to the hatch-contour interface.

The relationship between IHTT and void formation is confirmed across various toolpaths. Finally, the basis for a statistical frameworks, which predicts void occurrence as a function of component geometry and toolpath, as well as which can programmatically generate toolpaths that produce components with less near surface porosity, is presented. Future work includes a propose experiment that details different strategies to mitigate the occurrence of these near-surface voids, to determine the mitigation method which is most effective and efficient, as well as determine the physical mechanism of these near surface voids.

Chapter 2

Experimental Procedure

Experimental Design

Three, 21 mm Ti-6Al-4V cubes were fabricated in a 3DSystems ProX-320 PBFAM system, utilizing a 500 W fiber laser with an output wavelength of 1070 ± 10 nm. The parts were fabricated using default processing parameters and grade 23 Ti-6Al-4V powder with a size distribution of 15-45 microns, supplied by 3D Systems. As in [21], the argon flow was slightly reduced, with all other parameters remaining identical to optimal processing settings: a hatch speed of 1,250 mm/sec, a laser power of 245 W, and a layer thickness of 60 microns. Powder was spread with a flexible silicone recoater. Between each layer, the hatches were rotated by 23 degrees for the first 7 mm of build height, 47 degrees for the second 7 mm of build height, and 69 degrees for the remainder of the build height, except for one sample with no rotation. Computer aided designs of all samples are shown in figure 2-1.

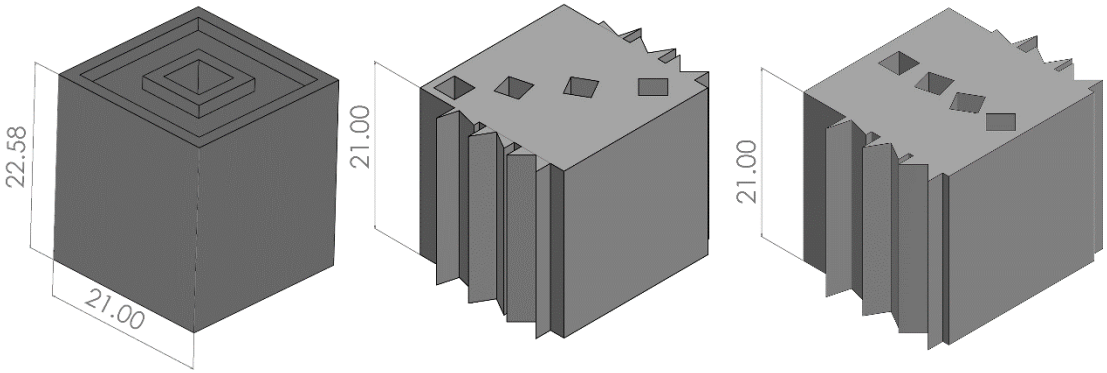

Figure **2-1**: (Left to right) Sample A, sample B, and sample C (all dimensions are in mm).

Each sample was designed with a different focus on short and long hatches. Sample A was designed so that cube contained primarily long hatches, except for the top perimeter which contained primarily shorter hatches. Sample B was designed so that it contained primarily long hatches in the center region, but contained short hatches in the sharp corners protruding from two of its sides. Sample C has nearly the same geometry as sample B, but the toolpath does not rotate between layers. As the laser moves into and out of the sharp corners (see chapter 3), this sample contains only long hatches.

CT Data Processing

One means to assess part quality and detect voids is through computed tomography (CT). This is a 3 dimensional imaging technique, during which a sample is irradiated with an X-ray beam, and the absorption of various parts of the sample is measured on a detector screen, while the sample is rotated. From this, a 3 dimensional model of the sample is reconstructed. CT scanning can be used to nondestructively detect voids and their morphologies within a sample, with a typical resolution as low as 5 microns [22] .

Each cube was CT scanned in a GE phoenix v|tome|x m 300, using a 300 kV micro-focus x-ray source to produce scans with a 15 micron voxel size. The location of voids (anomalous voxels) was automatically detected and registered to the build plate domain via software as described by [23], [24]. Raw CT data and extracted voids for sample B are shown in figure 2-2.

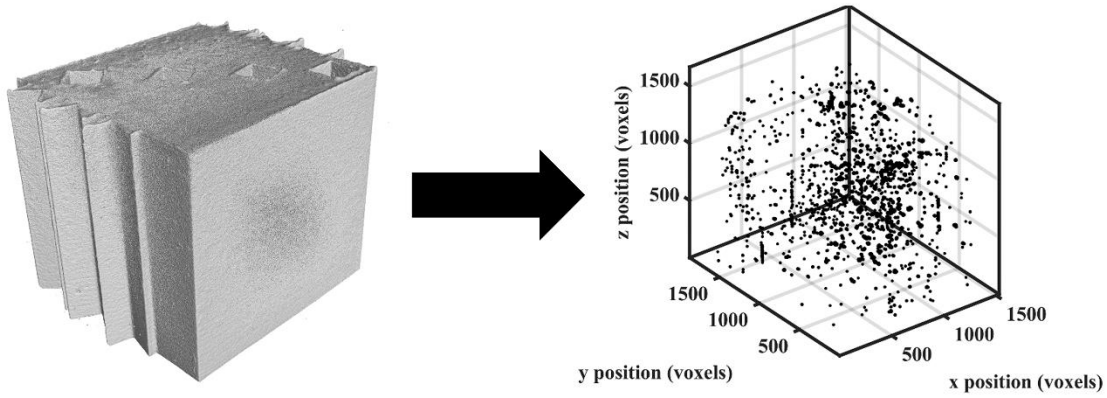

Figure **2-2**: Raw CT data of a component (left) and location of the voids within (right)

Each automatically-detected void was manually verified to minimize false positives. For this work, which focuses on near surface voids, only voids within 400 microns from the surface of each sample were used in subsequent analysis.

Toolpath Analysis

The machine's toolpath (laser power, speed, location, and trajectory associated with each laser strike) was provided by 3D Systems and verified using an in situ using a data acquisition system similar to that described in [25], [26]. For each void, the nearest intersection of the hatch and contour on the toolpath was located. The travel time along the hatching path, forward in time, was measured until the hatch path intersected with the same contour again. If the travel distance was merely the distance between the two points directly, i.e., a laser "jump" occurring at turnaround, the laser path was followed again until the next intersection of the contour. This quantity of time was defined as $t_{Forward}$. The phenomenon of between hatch jumps can be seen in figure 2-3.

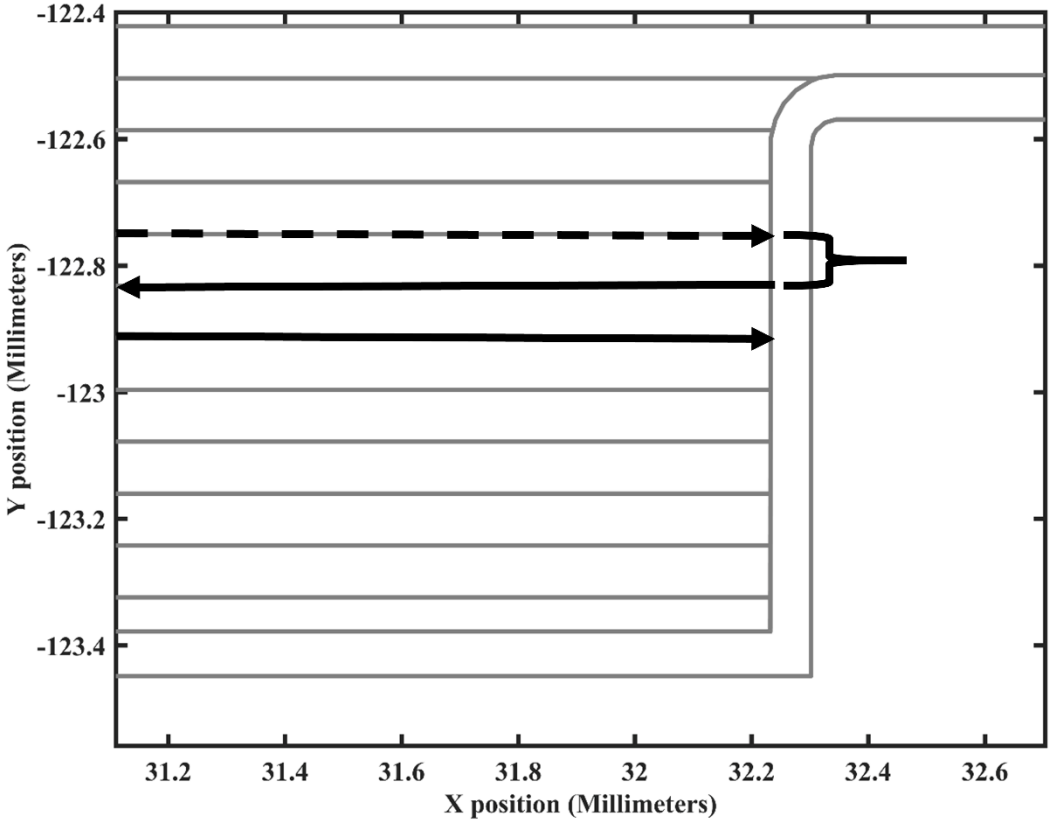

Figure **2-3**: Illustration of a jump between hatches. After completing the dashed hatch, the laser jumps the distance shown the by bracket.

In this example, the $t_{Forward}$ for the point at the end of the dashed hatch would be calculated as the time needed to complete both of the solid hatches to arrive at the next notable point on the contour. The time needed to go from the point at the end of the dashed hatch to the next point on the contour (the beginning of the first solid hatch) is trivial, as it is just the time needed for the laser to jump from one hatch to the next.

This process was repeated, measuring backwards in time, from the hatch-contour intersection nearest to the void. The travel time when scanning backwards was defined as t_{back}.

The inter-hatch travel time (IHTT) was defined as the lower of these two numbers associated with that void.

$$IHTT = min\,[t_{forward}, t_{back}]$$

The algorithm for determining IHTT is illustrated in figure 2-4. Any voxel in the CT space, whether a void or nominal voxel, can be mapped into the toolpath space, and has a corresponding IHTT.

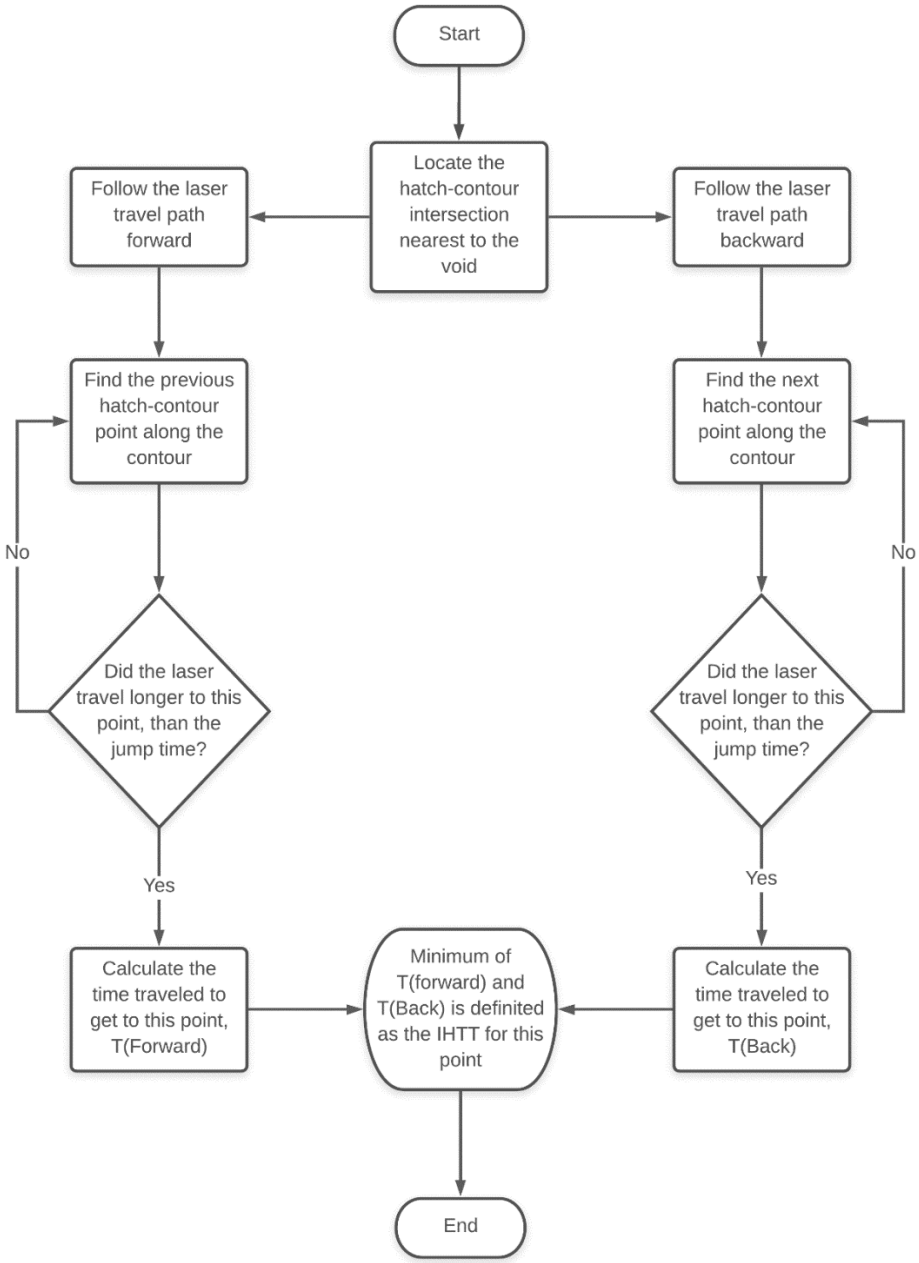

Figure **2-4**: Flow chart of IHTT calculation.

The inter-hatch travel time strongly depends part geometry. Voxels within a neighborhood of a thin or sharp features can be expected to—on average—have a shorter IHTT than voxels which are far away from such features. For instance, consider the voxel shown using

a solid grey square and highlighted by a dashed arrow in figure 2-5, which includes a portion of the laser toolpath for layer 300 of sample B. The path along which the laser travels forward to the neighboring contour point is shown using solid arrows, while the distance backwards in time is shown using dotted arrows. The IIHT for the voxel shown with a solid square is the minimum of the backward and forward distances multiplied by their respective travel speeds along each path (in this case, the time needed to travel forward). As shown in figure 2-5, points within thin geometries have smaller IHTTs than thick geometries.

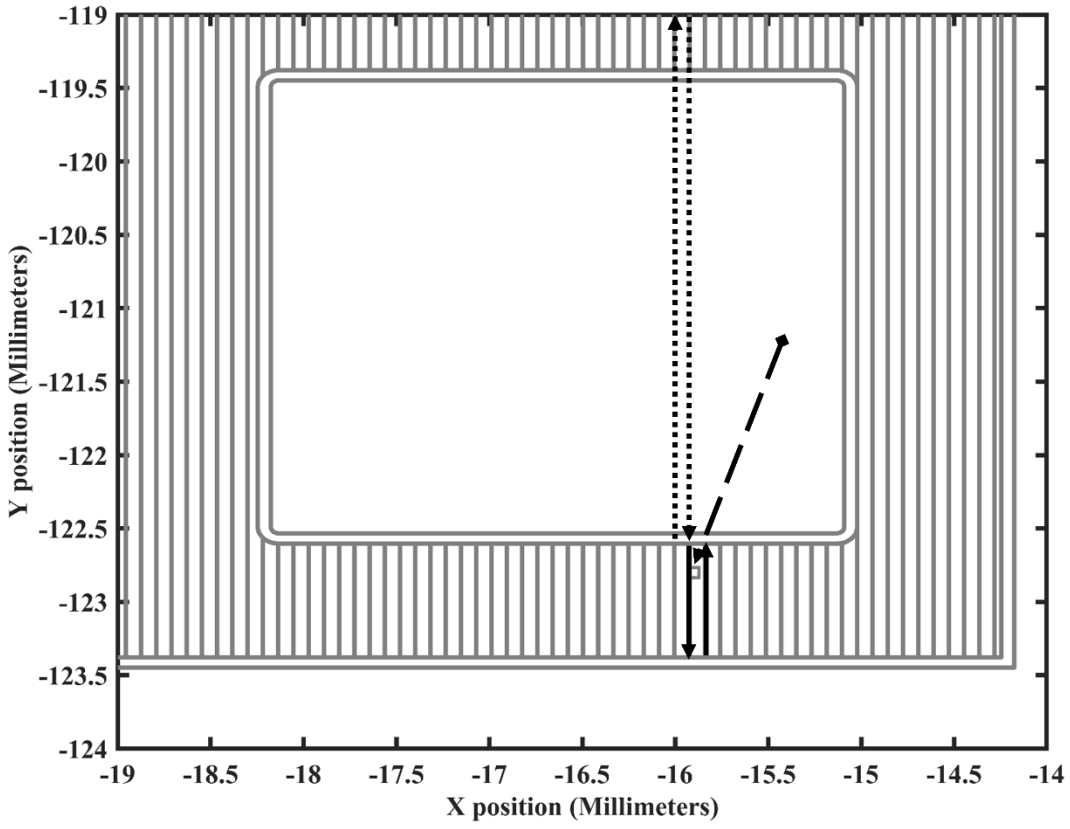

Figure **2-5**: Illustration of the short IHTT in a region of sample B with a thin geometry.

Similarly, voxels within sharp corners typically have smaller IHTTs than voxels in thick geometries. However, the IHTT at a given point may vary significantly with different scan directions. As illustrated in figure 2-6, showing vectors in layer 300 and 301 of sample B, a voxel located with a sharp feature (denoted by a square and highlighted by a red arrow) generally has a short IHTT except when (Figure 2-6 (b)) the angle of hatching is nearly parallel to one of the contours of the sharp feature.

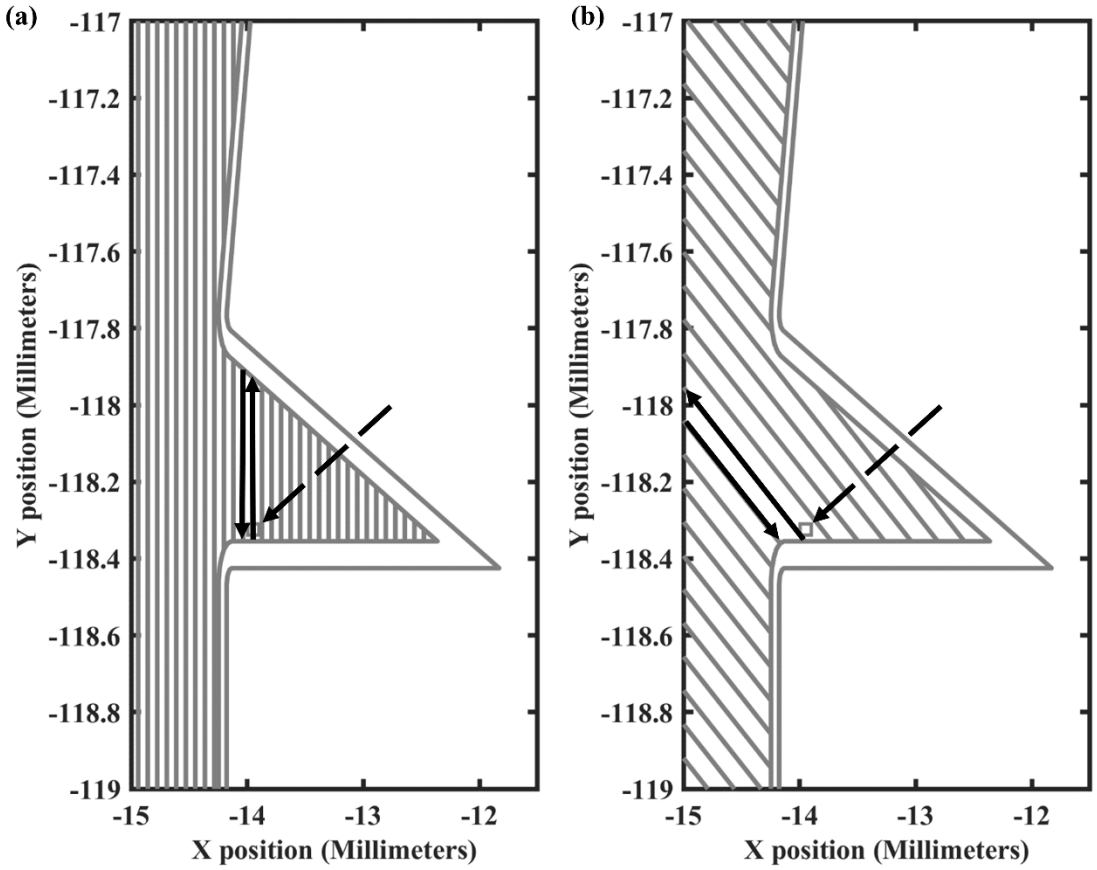

Figure **2-6**: Illustration of differing IHTT on layers 300 (left) and 301 (right) of sample B, in layer 301, the laser travels over the rest of the part after leaving the left of the figure before returning to the contour.

To test the hypothesis that IHTT is a statically-significant predictor of the likelihood of voids, IHTTs were computed for all voids (i.e. anomalous voxels) within 400 microns of the surface. A roughly balanced data set consisting of roughly 1400 nominal points per sample were also selected randomly from nominal voxels 200 microns from the surface. All nominal sampled were 200 microns from the surface, as that is the midpoint of the range of the voids' distances (0-400). The means of IHTT of nominal and anomalous voxel populations, as well the IHTT distribution for each population, were compared. To compare the mean IHTT of each population, a two sample t-test was performed for each sample. To compare the population distributions, probability histograms of nominal and anomalous IHTTs were juxtaposed.

Chapter 3

Results and Discussion

Near-surface voids per sample

Based on CT analysis of Samples A, B, and C, near surface voids were on average small (~60 microns in diameter) and appear spherical at the CT resolution (15 microns/voxel) utilized. Table 1 shows that sample B contains the greatest number of near surface voids, and sample C contains the fewest. This is significant because the two samples have the same features and cross-sectional geometries. In addition, the mean and median size of near surface voids in sample C is less than that of the near surface voids in sample B.

Table **3-1**: Number and diameter of near surface voids per sample.

Sample	Number of near surface voids	Mean diameter (microns)	Median diameter (microns)	Min diameter (microns)	Max diameter (microns)
A	234	65.7	54.8	18.6	345
B	481	63.7	50.5	18.6	323
C	76	47.1	36.4	18.6	142

Figure 3-1 shows typical near surface voids: spherical (at this resolution) and small.

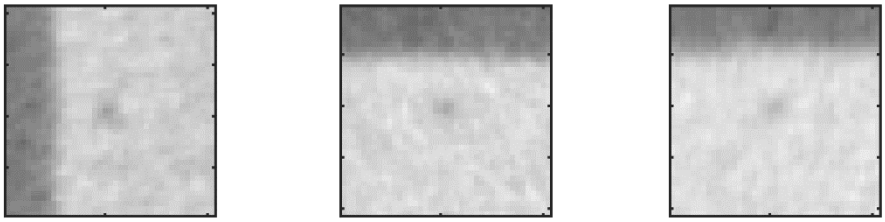

Figure **3-1**: CT images of representative near surface voids found in this work, taken from sample B.

Thin geometries (Sample A)

Figure 3-2 shows an isometric wireframe and a top-down view of sample A, which contains a thin perimeter atop it.

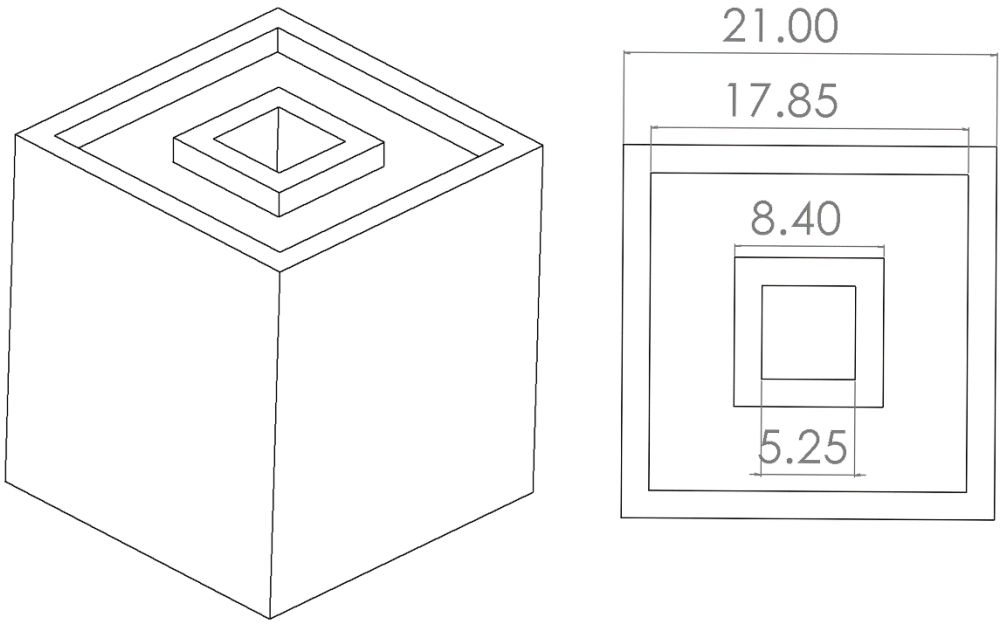

Figure **3-2**: Isometric wireframe (left) and top-down wireframe (right) view of CAD of sample A, a cube with a thin perimeter wall printed atop. All dimensions are in millimeters.

To investigate the influence of thin features on IHTT and near-surface voids, sample A was design to consist of solid block base atop which thin internal and external perimeter walls were built. As shown in Figure 3-2, sample A had a total height of 25.5 mm (1,700 CT voxels tall). The bottom 4.5 mm (300 CT voxels tall) of the build correspond to material that was lost

during removal of the part from the build plate. The solid block base extended to a height of 24 mm (CT voxels with a Z value from 300-1,600. The remaining height of 24.0-25.5 mm (1,600-1,700 CT voxels high) correspond to the thin internal and external perimeter, shown in red in figure 3-3.

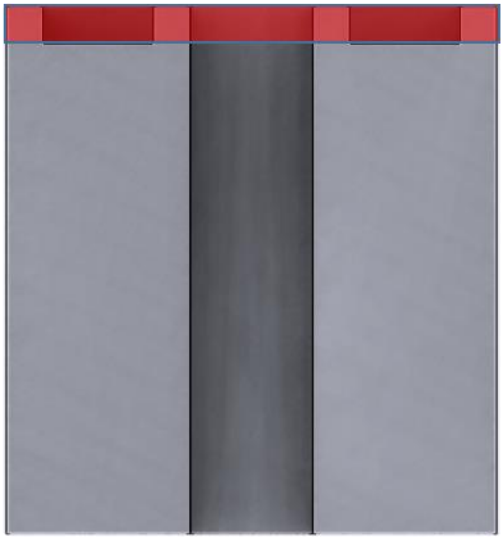

Figure **3-3:** Thin upper rim (highlighted in red) atop sample A, CAD cross-section.

More voids were found in the thin internal and external perimeter regions of sample A, than in any other similarly 1.5 mm tall (100 CT voxel height) region. Note that this analysis considers all voids, not just those near the surface. Figure 3-4 shows a histogram of void occurrence as a function of part height, measured in voxels. The thin walls at the top of the part have 85% more voids than the average of similarly tall sections within the solid block region of the part. Below a build height of 24.0 mm (1,600 CT voxels high), within the solid block region, an average of 49 voids (standard deviation of 7.95) were found per 1.5 mm (100 CT voxel) tall section. This is in contrast to the 91 voids within the thin walled region, 5.28 standard deviations above the mean of solid section.

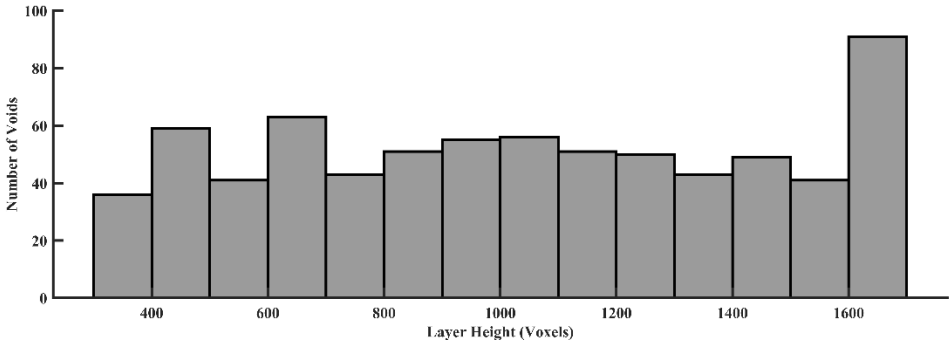

Figure **3-4**: Total void count as a function of height in sample A.

This result is striking given that data in figure 3-4 do not account for the fact that the thin walled region has a (60%) smaller cross sectional area than the rest of the cube. When normalized for cross-sectional area (number of voids in each 1.5 mm tall region divided by cross-sectional area), the thin-walled regions are shown to have 364% more voids than the average of the other sections, given the average number of voids per mm^2, 0.1187 (standard deviation of 0.0192) in the body of the cube. The number of voids in the ring, when normalized for cross-sectional area, is 22.48 standard deviations above the mean of the other sections, as shown in figure 3-5.

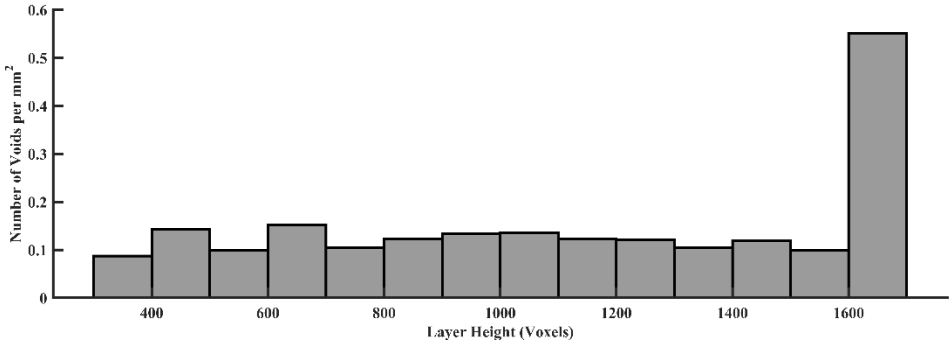

Figure **3-5**: Void count in sample A by height, normalized by cross-sectional area.

Given the large number of voids identified, the thin-walled region of sample A provides a good case to test the ability of IHTT to predict near-surface voids. Additionally, this case provides a wide range of IHTTs—see the nominal distribution in figure 3-6. If IHTT has no effect on void formation, we would expect a similar distribution for void and nominal voxels across all IHTTs. That is, if void formation is simply a byproduct of cross-sectional geometry, independent of toolpath, then both void and nominal population distributions ought to be independent of IHTT.

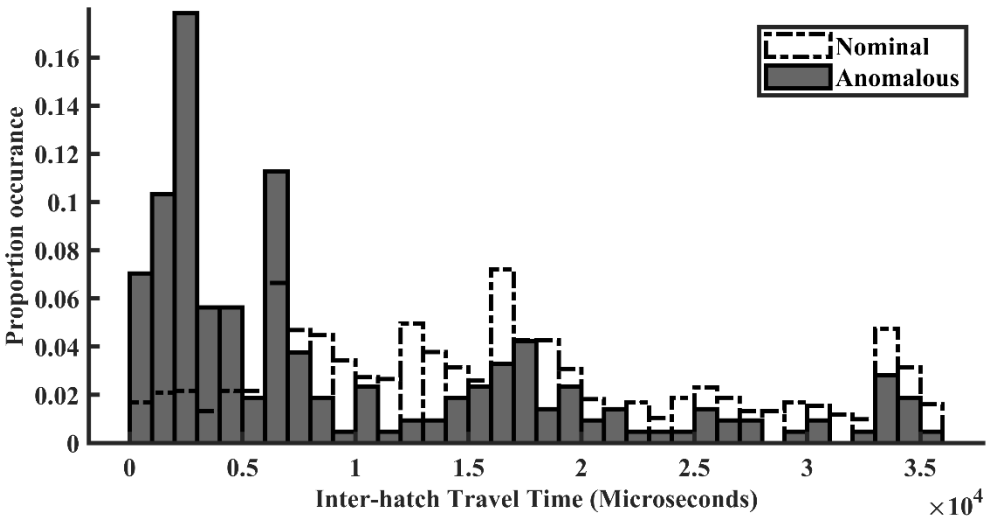

Figure **3-6**: Probability histograms of anomalous and nominal voxel populations in sample A.

The probability of a void occurrence was found to be significantly higher for voxels with short IHTTs. Normalized histograms (i.e. probability histograms) of void and nominal voxels as a function of IHTT are shown in Figure 3-6. Each distribution was normalized to enable comparison on the non-equal populations (213 nominal vs 1430 void voxels). The mean IHTT of anomalous voxels in sample A was 9,548 microseconds, while the mean IHTT of nominal voxels

was 16,814 microseconds, showing that shorter hatches are more likely to form near surface voids than long hatches ($p = 8.54*10^{-20}$).

Dividing the anomalous probability by the nominal probability for each bin of the histogram provides a representation ratio for voids, defined as the probability of a voxel with a given IHTT of being anomalous (part of a void) compared to the likelihood of a random voxel being anomalous. The representative ratio of voids is provided in figure 3-7. The representation ratio demonstrates that, for IHTTs less than 4,000 microseconds, a near-surface voxel is ~4 to 8 times more likely to be a void than a typical near surface voxel.

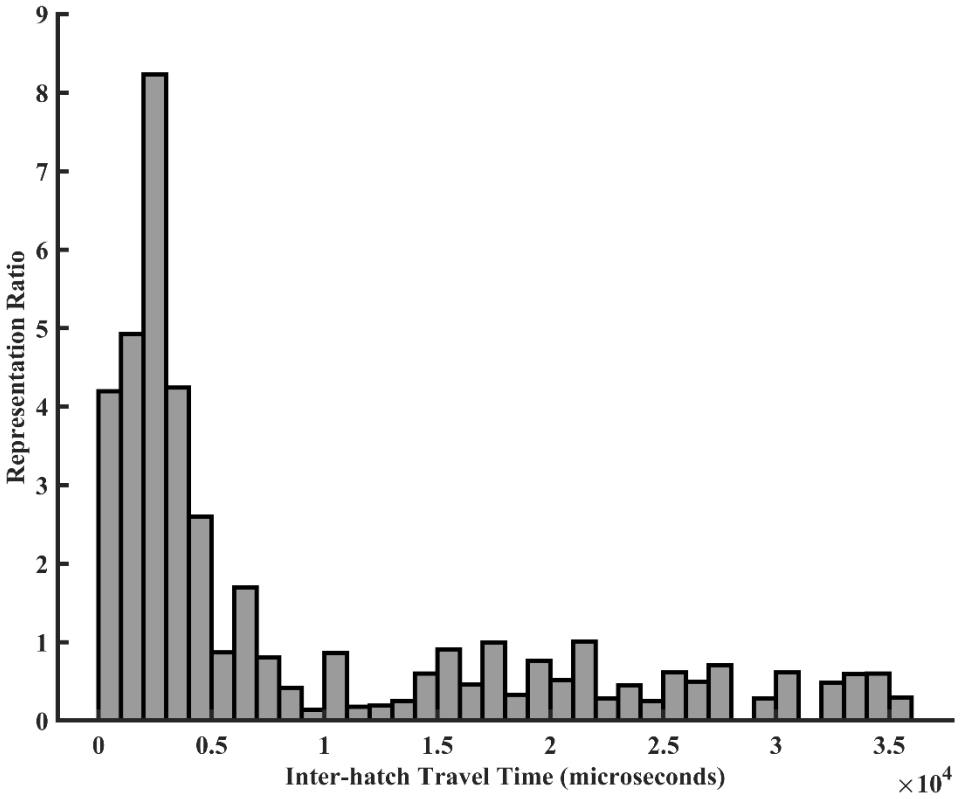

Figure **3-7**: Representation ratio of anomalous to nominal voxels in sample A

Sharp Features (Sample B)

Sample B was designed to investigate the effect of sharp features on IHTT and void formation, as with thin features, voxels near sharp features are more likely to have small IHTTs than other voxels. Figure 3-8 shows an isometric wireframe and a top-down view of sample B, which contains many sharp corners.

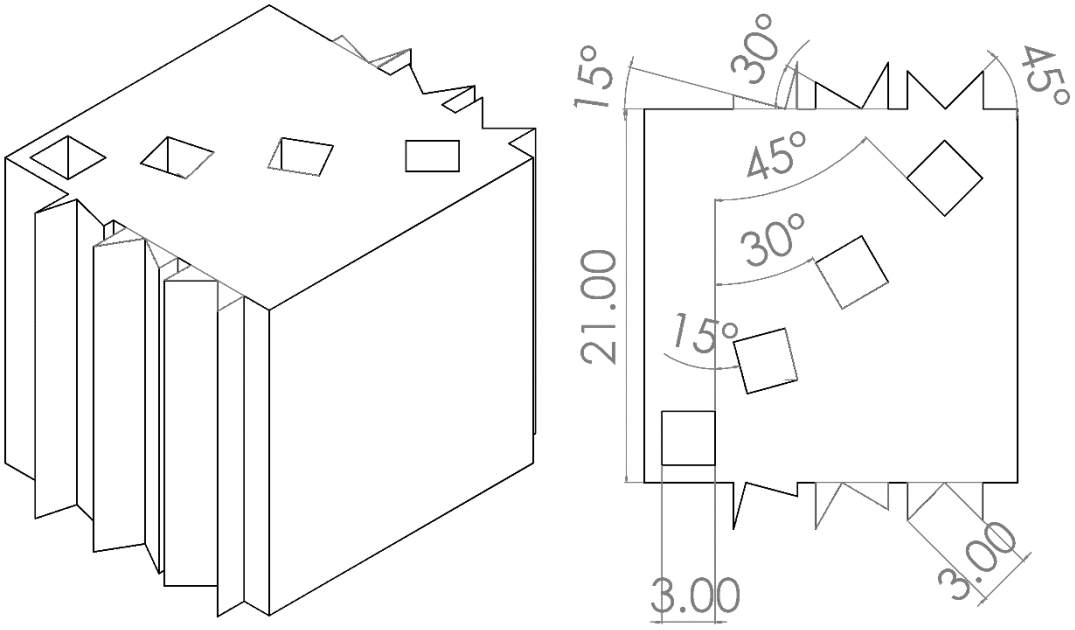

Figure **3-8**: Isometric wireframe (left) and top-down wireframe (right) view of CAD of sample B. All linear dimensions are in millimeters.

If, the results found in sample A—that the likelihood of voids is higher in regions with lower IHTTs—holds, then we should expect more near surface voids in thin geometries, such as

the thin regions (for example, illustrated by the dashed arrow in the upper left corner in figure 3-9 and near the many sharp protuberances along its surface).

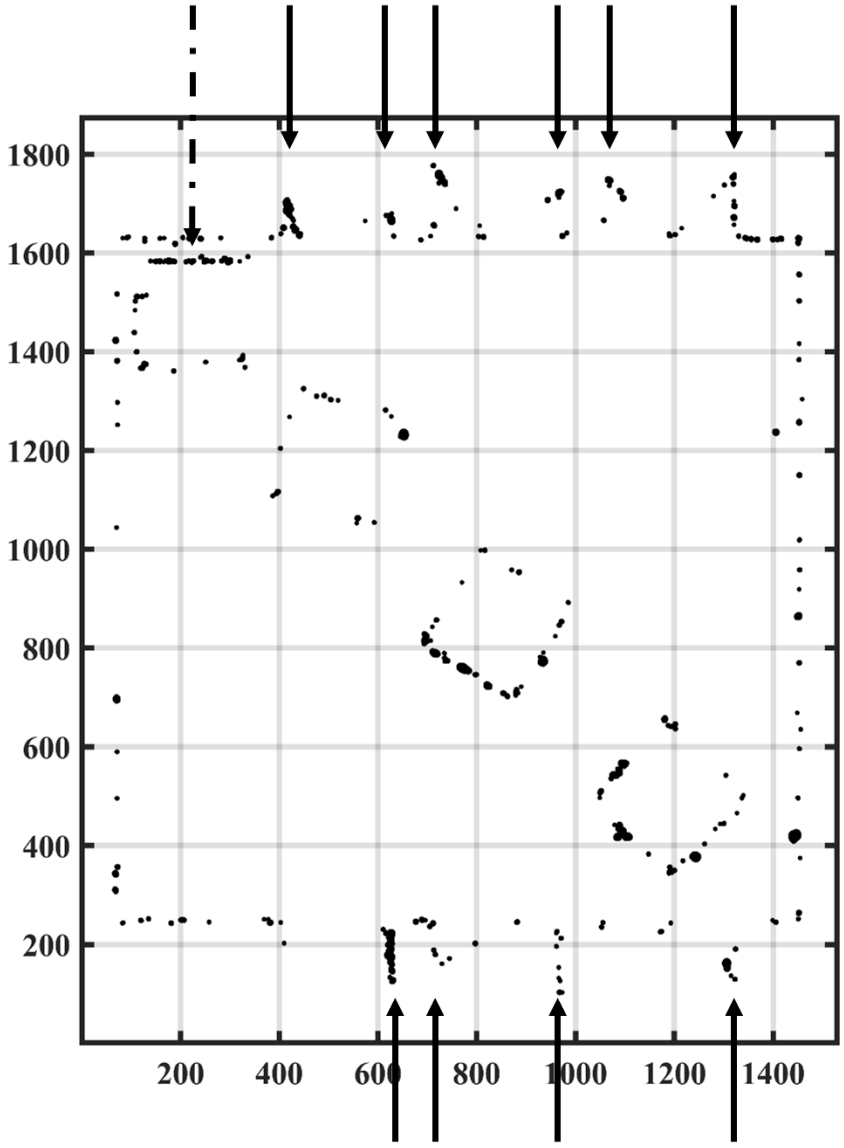

Figure **3-9**: Top down view of the positions of all near surface voids in sample B. Surface with low IHTT, thin geometry (solid arrows), surface with low IHTT, sharp corner (dashed arrow).

A top-down projection of all of the near surface voids via CT in sample B (figure 3-9) qualitatively confirms this hypothesis. Figure 3-9 shows a large concentration of voids in the thin region shown by the solid arrows and the sharp corner shown by the dashed arrow. Each of these regions have shorter IHTTs compared to other near-surface voxels within the sample. This effect is particularly apparent in the bottom and right edges of the hollow region highlighted by dashed arrow in figure 3-9—these edges larger IHTTs on average and almost no voids are found within.

A quantitative analysis of sample B showed that near-surface voxels with short IHTTs are statistically more likely to be voids. The mean IHTT of anomalous voxels in sample B was 9,709 microseconds. This is lower than the mean IHTT of randomly-selected nominal voxels of 12,294 microseconds to a statically significant extent ($p = 5.76 \times 10^{-6}$).

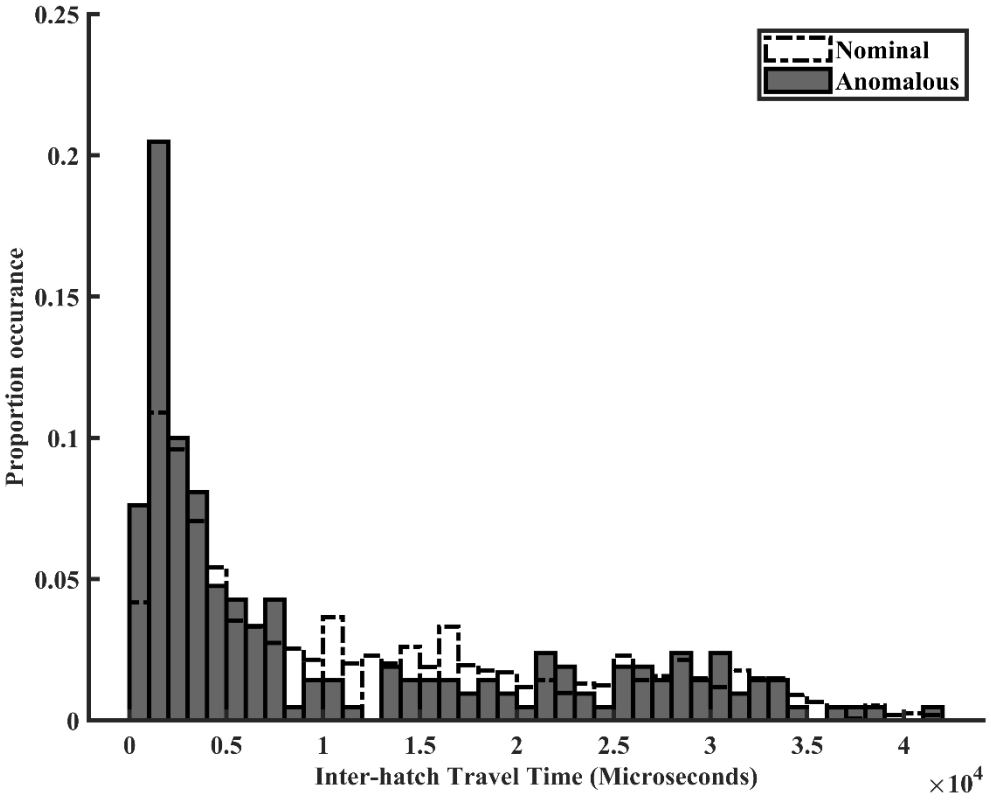

Figure **3-10**: Probability histograms of anomalous and nominal voxel populations in sample B.

As with sample A, for IHTTs less than 2,000 microseconds, a near-surface voxel is significantly more likely (~1.5x for sample B) to be a void than a typical near surface voxel.

Sharp Features with increased IHTT (Sample C)

To further test if near-surface voids were more strongly related to IHTT than to overall part geometry or cross-sectional area, sample C (figure 3-11) was designed to preserve sample B's cross-sectional area and sharp features but to significantly increase IHTTs. By requiring hatching in a single direction (parallel to horizontal axis) and moving hollow geometries to the middle of the sample, all short hatches, with travel times under 5,000 microseconds, were removed.

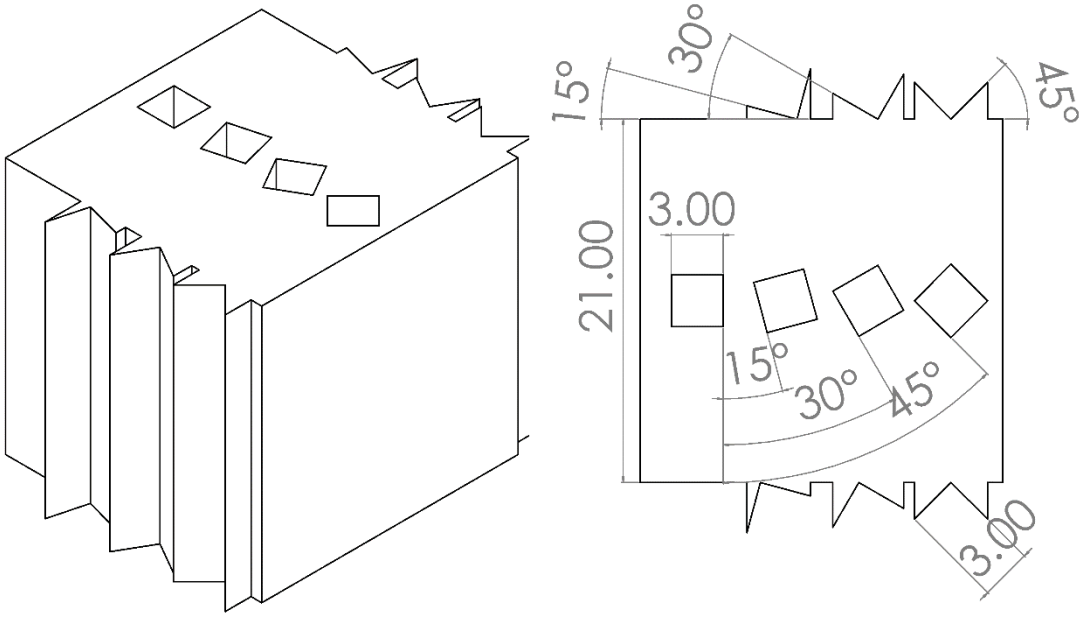

Figure **3-11**: Isometric wireframe (left) and top-down wireframe (right) view of CAD of sample C. All linear dimensions are in millimeters.

Figure 3-12 illustrates how short hatches were avoided in sample C: all hatches oriented themselves into the sharp features, and a sample of the actual toolpath of layer 300 of sample C.

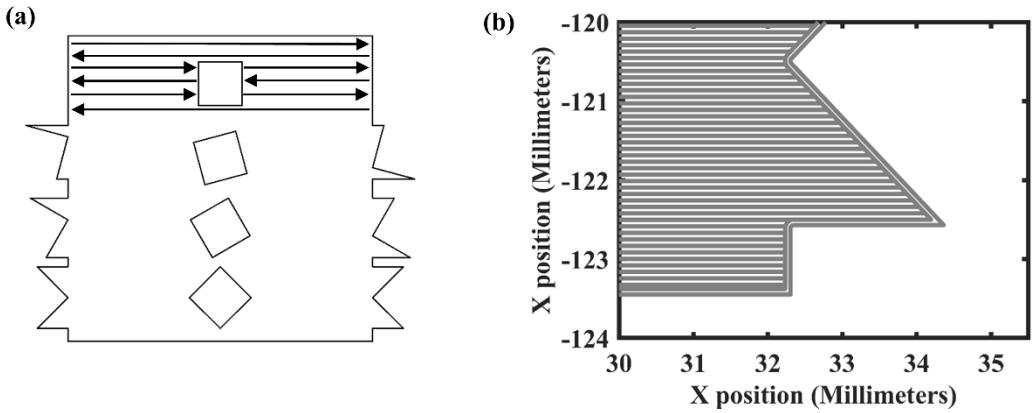

Figure **3-12**: (a) Illustration of sample C's hatching pattern, which was repeated on each layer. (b) Excerpt of actual toolpath on layer 300 of sample C.

As predicted, a large reduction in the number of near surface voids, compared to sample B, was observed. Sample B contained 447 near surface voids, whereas sample C contained only 74, a reduction of 83%. And while the two samples are not exactly identical, they have the same volume, total laser travel length (total energy input), cross-sectional area per layer, and features.

This reduction in near-surface voids strongly supports our hypothesis that short hatches are more likely to form near surface voids. However, statistical analysis of sample C is a difficult.

There are so few voids per bin on average that random noise plays a significant factor in bin count. The mean IHTT of anomalous voxels in sample C, 17,701 microseconds, is now higher than the IHTT of nominal voxels, 15,691 microseconds to just above a statistically significant extent ($p = 0.017$). This is likely because other factors are influencing the formation of voids. For example, the points around the edge of the part have higher IHTTs than the points around the central hollow features, but also have sharp, convex corners. This may point to an effect local geometries on the likelihood of near-surface void formation when IHTTs are sufficiently long. The cases studied in this work suggest an IHTT threshold value around 5,000 - 10,000 microseconds.

Recent findings by Schwalbach et al. [12], which showed that a series of short hatches is more likely pass over a contour-hatch interface that has not completely solidified from a prior laser pass. They further argue that such interactions result in variations in track widths and melt depths resulting from interactions of a large melt pool formed by multiple hatches. Such interactions are suggested to possibly lead to a greater likelihood of void formation. Our results confirm that regions nearer to the hatch contour interface are more likely to form voids, particularly near short hatches.

Chapter 4

Conclusions

This work introduced and identified the travel time between adjacent hatch strikes (IHTT) to be a significant indicator of the likelihood of near surface void formation. Using build plan data together with registered CT scan data of samples designed to exhibit varying IHTTs, features, and cross-sectional areas the following conclusions were drawn:

- Near-surface void formation is not a simply a byproduct of cross-sectional geometry; it is dependent on toolpath and a strong function of IHTT.
- The number of flaws formed in a component can vary greatly due to the toolpath used to construct it. Over thin-walled regions (Sample A) near-surface voids were ~4-8 times more likely to occur with IHTT less than 4,000 microseconds.
- Over sharp features (Sample B) near-surface voids were ~1.5 times more likely to occur with IHTT less than 2,000 microseconds.
- Sample geometry and hatching strategy can be modified (Sample C), without necessarily affecting cross-sectional area or total energy input, to significantly reduce near-surface voids by increasing IHTT for all hatches. For the samples and processing conditions considered in this work, a threshold IHTT on the order of 5,000 to 10,000 microseconds is estimated to significantly reduce near-surface voids.
- We suggest that regions with short IHTTs may be more likely to form voids as a consequence of the interactions of melt pools formed by neighboring hatches. Recent work of Schwalbach et al. [12] shows short that short, sequential hatches likely produce interact with a contour-hatch interface remains molten over multiple passes. The high-speed imaging data of Nassar et al [11] also show this phenomena, shorter hatches

solidifying after the next hatch ends, give an opportunity for melt pools from neighboring hatches to interact. It is possible that such interactions over already-molten material results in coalesce of the melt due to surface tension, forming a large volume which may either pull material from neighboring regions or affect subsequent layers. It is also possible that this repeated heating results in transition to keyhole formation [7], [10], [27].

To reduce the number of near-surface voids, we suggest modifications of the part geometry and/or hatching strategy to reduce the number of hatch points with short IHTTs (<5,000 – 10,000 microseconds). This approach presented here thus lays the groundwork for algorithmically-generated toolpaths which minimize the formation of near surface voids. Alternate approaches may involve feedforward control [28] of processing parameters to control melting and solidification. Ongoing work seeks to implement feed-forward control strategies and to identify other parameters which strongly correlate with near surface void formation, in order to produce a statistical model that to determine the number of flaws expected for a given build plan.

Chapter 5

Future Work

Future work is planned to determine the physical cause of these near surface voids. The discrepancy in void count between samples B and C shows that varying the toolpath can greatly decrease void occurrence. Understanding the physical mechanism of void formation could be used to develop toolpaths which reduce void occurrence while minimally impacting other manufacturing considerations, such as build time.

A series of cones are to be built with varying thickness, so that this effect can be measured over a continuum of feature sizes. Additional sections of toolpath may be added, and toolpath speeds may be changed, in order to determine the physical mechanism of flaw formation.

The basic cone design is shown in figure 5-1.

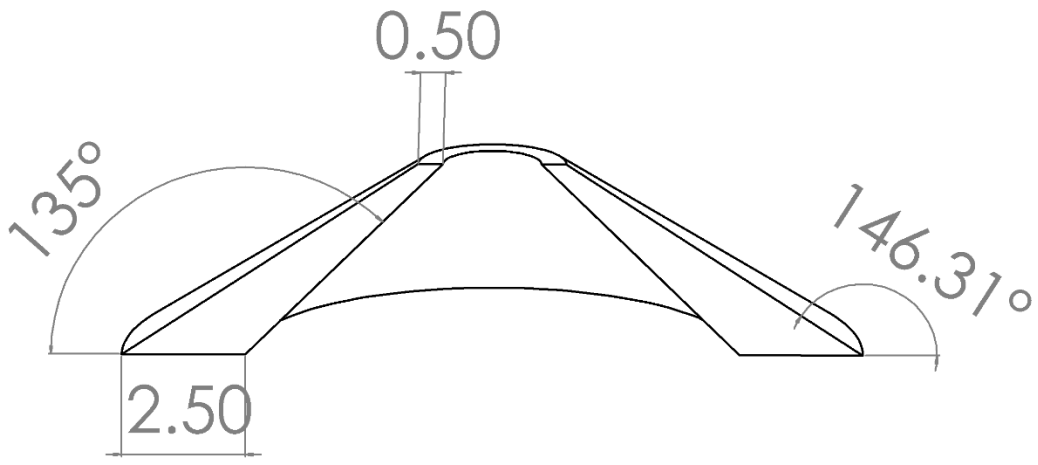

Figure **5-1**: Cross section of a sample cone. All linear dimensions are in millimeters.

The wall thickness of 2.5 mm on the bottom of the cone tapers to 0.5 mm over a height of 4 mm. The inner dimeter starts at 10 mm and tapers to 2 mm. The IHTT of points on the outer surface decreases as a function of height due to the laser traveling over less build material, as well as traveling over a shorter jump. The IHTT of points along the inner surface decreases as a function of height only due to the laser traveling over less build material.

Sample 1 is this cone fabricated with all default parameters, to act as a control. Sample 2 we be built as the same cone, but the jump speed will be reduced to 3 m/s. This will allow for longer times between adjacent hatches. The IHTT of points on the outer surface will be larger than that of sample 1, but the IHTT of points on the inner surface will be identical to sample 1. However, allowing a pause as the laser jumps across the part will allow the thin geometry to cool. If short IHTTs cause voids due to regions with short hatches getting progressively hotter, leading to keyholes, this sample will have fewer voids than sample 1. However, if the cause of these near surface voids lies in the fluid dynamics of molten metal pooling between adjacent hatches, the number of near surface voids along the inner surface will likely be unchanged from sample 1.

Sample 3 is the same geometry as sample 1 and 2, with the addition of an inner cone and outer sheath.

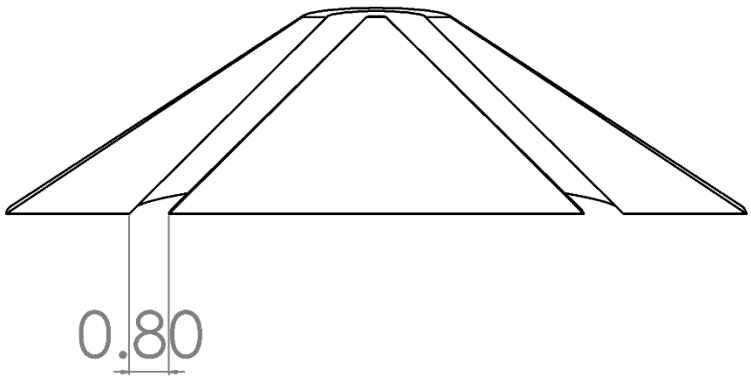

Figure **5-2**: Cross section of sample 3. All dimensions are in millimeters.

The horizontal distance between the sample cone and the inner cone is 0.8 mm. The jump speed will be set to 3 m/s, to match sample 2. All other build parameters will be the default. On each layer, the hollow cone is fabricated first and then the inner solid cone. The IHTT of points along the inner and outer cones will be equal to those of sample 2. While the laser is imputing more power into the sample, the additional objects are designed to act as thermal supports, pulling heat away from the thin region as it is lased. If short IHTTs cause voids due to regions with short hatches getting progressively hotter, leading to keyholes, this sample will have fewer voids than sample 1 and 2, as the thermal support will be pulling heat away from the thin region faster than for samples 1 and 2. However, if the cause of these near surface voids lies in the fluid dynamics of molten metal pooling between adjacent hatches, the number of near surface voids along the inner surface will likely be unchanged from samples 1 and 2.

Sample 4 is designed to further investigate the effect of slowing jump speed to reduce IHTT. It is the same geometry as sample 1 and 2, but the jump speed will be decreased to 1.25 m/s, The IHTT of the points along the outer surface will be further reduced, without changing the IHTT of the points along the inner surface. Allowing a pause as the laser jumps across the part will allow the thin geometry to cool. If short IHTTs cause voids due to regions with short hatches getting progressively hotter, leading to keyholes, this sample will have fewer voids than samples 1, 2 and 3. However, if the cause of these near surface voids lies in the fluid dynamics of molten metal pooling between adjacent hatches, the number of near surface voids along the inner surface will likely be unchanged from samples 1, 2, and 3.

Sample 5 does not aim to determine the mechanism of void formation at all, but instead aims to fix the voids as they form with as little delay as possible. It is the same hollow cone geometry as samples 1, 2, and 4, but with the addition of two post-contours. Both post contours will be offset so that they are further within the part than the default pre-contours, (which will still

be included in this component). The 2 additional post contours will be offset from each other. If the voids that form near the surface tend to form near to the surface of the layer, opposed to forming below the layer being fabricated, remelting this region may smooth the surface and heal the voids.

Together with the conclusions based on this current work, the proposed work will further the conclusion that short inter-hatch travel times negatively influence the likelihood of near surface flaw formation. Additionally, insight into the physical mechanism behind the formation of these near surface flaws will be gained. Understanding the physical mechanism may allow researchers to better define the critical IHTT cutoff as well as extend this knowledge to other material systems.

The proposed work will likely improve the manufacturability of thin geometries by L-PBF. We propose a method (post contours in sample 5) which may quickly mitigate the presence of near surface voids. Even if this method does not prove effective, samples 1-4 will display the effectiveness of other methods for preventing near surface void formation in thin geometries. As well, The knowledge of the physical mechanism will lay the groundwork for additional mitigation mechanisms.

References

[1] S. Tammas-Williams, P. J. Withers, I. Todd, and P. B. Prangnell, "The Influence of Porosity on Fatigue Crack Initiation in Additively Manufactured Titanium Components," *Sci. Rep.*, vol. 7, no. 1, p. 7308, Dec. 2017, doi: 10.1038/s41598-017-06504-5.

[2] B. Cheng, S. Shrestha, and K. Chou, "Stress and deformation evaluations of scanning strategy effect in selective laser melting," *Addit. Manuf.*, vol. 12, pp. 240–251, Oct. 2016, doi: 10.1016/j.addma.2016.05.007.

[3] S. Tammas-Williams, H. Zhao, F. Léonard, F. Derguti, I. Todd, and P. B. Prangnell, "XCT analysis of the influence of melt strategies on defect population in Ti–6Al–4V components manufactured by Selective Electron Beam Melting," *Mater. Charact.*, vol. 102, pp. 47–61, Apr. 2015, doi: 10.1016/j.matchar.2015.02.008.

[4] W. E. Frazier, "Metal Additive Manufacturing: A Review," *J. Mater. Eng. Perform.*, vol. 23, no. 6, pp. 1917–1928, Jun. 2014, doi: 10.1007/s11665-014-0958-z.

[5] T. DebRoy *et al.*, "Additive manufacturing of metallic components – Process, structure and properties," *Prog. Mater. Sci.*, vol. 92, pp. 112–224, Mar. 2018, doi: 10.1016/j.pmatsci.2017.10.001.

[6] "Additive Manufacturing of Metallic Materials MATSE 567 Beam Material Interactions in Additive Manufacturing," Steidle 401 Penn State, 04-Feb-2020.

[7] R. Cunningham *et al.*, "Keyhole threshold and morphology in laser melting revealed by ultrahigh-speed x-ray imaging," *Science*, vol. 363, no. 6429, pp. 849–852, Feb. 2019, doi: 10.1126/science.aav4687.

[8] A. A. Martin *et al.*, "Dynamics of pore formation during laser powder bed fusion additive manufacturing," *Nat. Commun.*, vol. 10, no. 1, p. 1987, Dec. 2019, doi: 10.1038/s41467-019-10009-2.

[9] M. A. Groeber, E. Schwalbach, S. Donegan, K. Chaput, T. Butler, and J. Miller, "Application of characterization, modelling, and analytics towards understanding process-structure linkages in metallic 3D printing," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 219, p. 012002, Jul. 2017, doi: 10.1088/1757-899X/219/1/012002.

[10] W. E. King *et al.*, "Observation of keyhole-mode laser melting in laser powder-bed fusion additive manufacturing," *J. Mater. Process. Technol.*, vol. 214, no. 12, pp. 2915–2925, Dec. 2014, doi: 10.1016/j.jmatprotec.2014.06.005.

[11] A. R. Nassar, M. A. Gundermann, E. W. Reutzel, P. Guerrier, M. H. Krane, and M. J. Weldon, "Formation processes for large ejecta and interactions with melt pool formation in powder bed fusion additive manufacturing," *Sci. Rep.*, vol. 9, no. 1, p. 5038, Dec. 2019, doi: 10.1038/s41598-019-41415-7.

[12] E. J. Schwalbach, S. P. Donegan, M. G. Chapman, K. J. Chaput, and M. A. Grober, "A discrete source model of powder bed fusion additive manufacturing thermal history," *Addit. Manuf.*, vol. 25, pp. 485–498, Jan. 2019, doi: https://doi.org/10.1016/j.addma.2018.12.004.

[13] J. R. Araujo, J. J. Rodriguez-Andina, J. Farina, F. Vidal, J. L. Mato, and M. A. Montealegre, "FPGA-based laser cladding system with increased robustness to optical defects," in *IECON 2012 - 38th Annual Conference on IEEE Industrial Electronics Society*, Montreal, QC, Canada, 2012, pp. 4688–4693, doi: 10.1109/IECON.2012.6389491.

[14] G. Bi, B. Schürmann, A. Gasser, K. Wissenbach, and R. Poprawe, “Development and qualification of a novel laser-cladding head with integrated sensors,” *Int. J. Mach. Tools Manuf.*, vol. 47, no. 3–4, pp. 555–561, Mar. 2007, doi: 10.1016/j.ijmachtools.2006.05.010.
[15] G. Bi, A. Gasser, K. Wissenbach, A. Drenker, and R. Poprawe, “Identification and qualification of temperature signal for monitoring and control in laser cladding,” *Opt. Lasers Eng.*, vol. 44, no. 12, pp. 1348–1359, Dec. 2006, doi: 10.1016/j.optlaseng.2006.01.009.
[16] Q. Wang, P. (Pan) Michaleris, A. R. Nassar, J. E. Irwin, Y. Ren, and C. B. Stutzman, “Model-based feedforward control of laser powder bed fusion additive manufacturing,” *Addit. Manuf.*, vol. 31, p. 100985, Jan. 2020, doi: 10.1016/j.addma.2019.100985.
[17] A. R. Nassar, J. S. Keist, E. W. Reutzel, and T. J. Spurgeon, “Intra-layer closed-loop control of build plan during directed energy additive manufacturing of Ti–6Al–4V,” *Addit. Manuf.*, vol. 6, pp. 39–52, Apr. 2015, doi: 10.1016/j.addma.2015.03.005.
[18] B. Zhang, Y. Li, and Q. Bai, “Defect Formation Mechanisms in Selective Laser Melting: A Review,” *Chin. J. Mech. Eng.*, pp. 1–13, 2017.
[19] S. Leuders *et al.*, “On the mechanical behaviour of titanium alloy TiAl6V4 manufactured by selective laser melting: Fatigue resistance and crack growth performance,” *Int. J. Fatigue*, vol. 48, pp. 300–307, Mar. 2013, doi: 10.1016/j.ijfatigue.2012.11.011.
[20] A. J. Sterling, B. Torries, N. Shamsaei, S. M. Thompson, and D. W. Seely, “Fatigue behavior and failure mechanisms of direct laser deposited Ti–6Al–4V,” *Mater. Sci. Eng. A*, vol. 655, pp. 100–112, Feb. 2016, doi: 10.1016/j.msea.2015.12.026.
[21] S. Coeck, M. Bisht, J. Plas, and F. Verbist, “Prediction of lack of fusion porosity in selective laser melting based on melt pool monitoring data,” *Addit. Manuf.*, vol. 25, pp. 347–356, Jan. 2019, doi: 10.1016/j.addma.2018.11.015.
[22] A. du Plessis, I. Yadroitsev, I. Yadroitsava, and S. G. Le Roux, “X-Ray Microcomputed Tomography in Additive Manufacturing: A Review of the Current Technology and Applications,” *3D Print. Addit. Manuf.*, vol. 5, no. 3, pp. 227–247, Sep. 2018, doi: 10.1089/3dp.2018.0060.
[23] C. Gobert, E. W. Reutzel, J. Petrich, A. R. Nassar, and S. Phoha, “Application of supervised machine learning for defect detection during metallic powder bed fusion additive manufacturing using high resolution imaging.,” *Addit. Manuf.*, vol. 21, pp. 517–528, May 2018, doi: 10.1016/j.addma.2018.04.005.
[24] J. Petrich, C. Gobert, S. Phoha, A. R. Nassar, and E. W. Reutzel, “Machine Learning for Defect Detection for PBFAM using High Resolution Layerwise Imaging Coupled with Post-Build CT Scans,” in *Proceedings of the 28th Annual International Solid Freeform Fabrication Symposium – An Additive Manufacturing Conference*, Austin, TX, 2017, pp. 1363–1381.
[25] A. J. Dunbar and A. R. Nassar, “Assessment of optical emission analysis for in-process monitoring of powder bed fusion additive manufacturing,” *Virtual Phys. Prototyp.*, vol. 13, no. 1, pp. 14–19, Jan. 2018, doi: 10.1080/17452759.2017.1392683.
[26] A. J. Dunbar, A. R. Nassar, E. W. Reutzel, and J. J. Bletcher, “A REAL-TIME COMMUNICATION ARCHITECTURE FOR METAL POWDER BED FUSION ADDITIVE MANUFACTURING,” presented at the 27th Annual International Solid Freeform Fabrication Symposium, Austin, TX, 2016.
[27] Q. Guo *et al.*, “Transient dynamics of powder spattering in laser powder bed fusion additive manufacturing process revealed by in-situ high-speed high-energy x-ray imaging,” *Acta Mater.*, vol. 151, pp. 169–180, Jun. 2018, doi: 10.1016/j.actamat.2018.03.036.
[28] D. W. Wang *et al.*, “Selective Laser Melting Under the Reactive Atmosphere: A Convenient and Efficient Approach to Fabricate Ultrahigh Strength Commercially Pure

Titanium Without Sacrificing Ductility," *SSRN Electron. J.*, 2019, doi: 10.2139/ssrn.3310270.

Appendix

IHTT calculation code

```
%Set bothCoords{1} equal to an N*3 matrix where the columns are the X, Y, Z
%coordinates of the N voids you wish to calculate the IHTT for

addpath('.../Utilities'); %must have Utilities folder with [redacted] function in path
voxelSz = 10 %Size of DICOM voxels, in microns
layerHght = 60; %Build layer height in microns
n_below = 20; %For nominal data sampling, number of voxels below the surface to sample nominals from
jump2scanRatio = 12; %Ratio of jump speed to scan speed. Output values are in effective scan length, i.e,
the amount of scan length that would take the same amount of time to traverse as needed to return to the
hatch-contour interface after departing from the point of interest

%% Section 1 This section loads the DICOMs. This is needed to undertake registration in Section X

DICOM_Path = [file path]; %editable
FileParseString = 'ct%05u.dcm'; %editable
[CTImageStack, Info] = LoadDicomStack(DICOM_Path);
clear CTImageStack
%% Sections 2 This section loads the DistMat. This is needed to check that your void is within your part

DistMat = load(file path); %editable
%% Section 3 This section contains the last of the editable fields

data = %redacted, proprietary methods, data extraction %editable
part_idx = 3; %This should agree with the part number inputted with the DICOM path!

contour_params = %Redacted, determines which part of proprietary data structure is contours
'cell2mat(data.parameters(:,8)) == 0' correponds to inner (CCW) contour, setting 0 to 1 will find the outer
(CW) contour instead
hatch_params =  %Redacted; proprietary method to determine which part of proprietary data structure is
hatches
start_params =  %Redacted; proprietary method to determine which part of proprietary data structure is the
beginning of hatches
end_params =  %Redacted; proprietary method to determine which part of proprietary data structure is the
end of hatches

extra_layers = 2; %currently unused; See block in section 5

real_vector_list = %Redacted; This block checks for vector cells with value '[]' and removes them from the
vector_list
```

```
remove_these =  %Redacted; proprietary method to determine which part of proprietary data structure is superfluous
real_vector_list(remove_these) = [];

nth_void = 0; %initializes process

%% Section 4 Creating the sample of points n voxels below the surface

[r,c,v] = ind2sub(size(DistMat.DistMat),find(DistMat.DistMat == -n_below));
void_coords = [r c v];
void_coords = datasample(void_coords,100000); %sample slightly more points than you want to select
deleteList = [];
for i = 1:length(void_coords)
   deleted = false;
   x = void_coords(i,1);
   y = void_coords(i,2);
   z = void_coords(i,3);
   location = getPatientPos([x,y,z], Info(z))*1000;
   location(3) = ceil(location(3)/layerHght);

   if location(3) > length(real_vector_list) %prevents selecting nominals above build vectors (Possible due to slight registration error)
      deleteList = [deleteList; i];
      deleted = true;
   end
   %for selecting nominal points from only a region
%    if (z>2161 | z<1928) & deleted == false
%       deleteList = [deleteList; i];
%       fprtinf('Note: You are selecting nominals from a constained set. Review Section 5 for details!')
%    end
end
void_coords(deleteList,:) = [];
void_coords = datasample(void_coords,5000); %It will take roughly 1 hour to process 50000 points on ARL 15468

bothCoords{2} = void_coords;


%% Section 5 Begins the analysis by point. Defines the layers of interest in each version
dist_list = [];
for ii = 1:2 %i = 1: real void coordinates , i = 2 sampled nominal points
   void_coords  = bothCoords{ii};
   %    for i = 1:length(void_coords)
   for i  = 1
      x = void_coords(i,1);
      y = void_coords(i,2);
      z = void_coords(i,3);

      nth_void = nth_void + 1; %counter

      distance_from_isosurface = -DistMat.DistMat(x,y,z); %By convention, negative distmat is inside part; here, we reserve this.
```

```
distance_from_isosurface = distance_from_isosurface * voxelSz %Voxels to microns
if distance_from_isosurface < 0
   warning('Void number %u is outside of the mask') %fix this
end

location = getPatientPos([x,y,z], Info(z))*1000; %translates the CT coordinate into build coordinates
location(3) = ceil(location(3)/layerHght); %Denominator is layer height in microns
if location(3) > 360
   location(3) = location(3) + 40;
end
if location(3) > length(real_vector_list) | location(3) < 1
   continue
end
% This section could be used to find the IHTT of coordinates a
% predetemined number of layers above/below the point of interest;
% development needed

% This block checks to see if the extra layers above/below the void correspond to actual vectors, and
removes them if not.
% %       for q = -extra_layers:extra_layers
% %          layers_of_interest(q+extra_layers+1) = location(3) + q;
% %       end
% %
% %       if (location(3) + extra_layers) > length(real_vector_list)
% %          layers_to_remove_above = length(real_vector_list) - location(3) + extra_layers;
% %          t = layers_of_interest(1:end-layers_to_remove_above);
% %       else
% %          layers_to_remove_above = 0;
% %       end
% %
% %       if (location(3) - extra_layers) < 1
% %          layers_to_remove_below = -(location(3) - extra_layers) + 1;
% %          layers_of_interest = layers_of_interest(layers_to_remove_below+1:end);
% %       else
% %          layers_to_remove_below  = 0;
% %       end

% Determines nearest hatch points


%       for h = 1:length(layers_of_interest) %Use if the block
%       above has been developed

overwriteForward = false;
overwriteVor = false;
hatch_idx = find(ismember(data.vectors{part_idx,location(3)}(3,:), hatch_params)); %detemines
which idx are hatches
hatch_pts = data.vectors{part_idx,location(3)}(1:2,hatch_idx); %finds the location of those idx
k = dsearchn(transpose(hatch_pts),transpose(location(1:2))); %Finds nearest hatch point in contour to
given point, returns as point1
point1 = hatch_pts(:,k);
%       end
point1_idx = hatch_idx(k);
```

```
        j = 2;
        distance = 1001;
        testPt = NaN;
        distance2HatchptList = [];
        while k+j < length(hatch_pts)+1 %j initalizes as 2, so the script does not check the point one ahead of
k.
            testPt = hatch_pts(:,k+j);
            distance = pdist([point1(1),point1(2);testPt(1),testPt(2)]);
            distance2HatchptList = [distance2HatchptList [j; distance; testPt]];
            j = j + 1;
        end
        if k < length(hatch_pts)-1
            leastDistance = min(distance2HatchptList(2,:));
            index = find(distance2HatchptList(2,:) == leastDistance);
            j = index + 1; %j is initalized at 2
            lastPt = distance2HatchptList(3:4,index);
            lastPtIdx = hatch_idx(k+j);
        end

        if k+j == length(hatch_pts) %condition triggered if no point is within dthreshold distance (this
shouldn't ever be true?)
            overwriteForward = true;
        end

        %creates a list of each hatch length
        vectorDistanceList = [];
        vecc = data.vectors{part_idx,location(3)};
        for qq = 1:length(data.vectors{part_idx,location(3)})-1 %list of distances between vector points is one
shorter than the number of points
            %vectorDistanceList(qq) = the distance from the qqth point to the qq+1thpoint, laser on, if the qqth
point is at the beginning of the hatch, NaN otherwise
            vectorDistanceList(qq) = pdist([vecc(1,qq),vecc(2,qq);vecc(1,qq+1),vecc(2,qq+1)]); %only finds
the values that are of travel with laser ON
        end

        for qq = 1:length(data.vectors{part_idx,location(3)})-2 %must be shorter because of vec format
            %vectorJumpDistanceList(qq) = distance from the qqth point to the qq+1th point, laser off, if the
qqth point is at the end of a hatch, NaN otherwise.
            vectorJumpDistanceList(qq) =
pdist([vecc(1,qq),vecc(2,qq);vecc(1,qq+2),vecc(2,qq+2)])/jump2scanRatio; %only finds the values that are
of travel with laser OFF
            %jump2scanRatio scales down the time spent in jumping to an effective scan distance
        end



        if point1_idx+3*j+1 > length(vectorDistanceList)
            overwriteForward = true;
        end
```

```
%determines number of hatches forward to count
if overwriteForward == false
   forwardDistNoJump = nansum(vectorDistanceList(point1_idx:lastPtIdx-1));
   forwardDist = forwardDistNoJump + nansum(vectorJumpDistanceList(point1_idx:lastPtIdx-1));
end

jj = 2;
distance2 = 1001;
distanceBack2HatchptList = [];
while k-jj > 0
   testPt2 = hatch_pts(:,k-jj);
   distance2 = pdist([point1(1),point1(2);testPt2(1),testPt2(2)]);
   distanceBack2HatchptList = [distanceBack2HatchptList [jj; distance2; testPt2]];
   jj = jj + 1;
end
if k > 2
   leastDistance2 = min(distanceBack2HatchptList(2,:));
   index2 = find(distanceBack2HatchptList(2,:) == leastDistance2);
   jj = index2 + 1; %j is 2 initalized
   firstPt = distanceBack2HatchptList(3:4,index2);
   firstPtIdx = hatch_idx(k-jj);
end
if k-jj < 1
   overwriteVor = true;
end

if overwriteVor == false
   vorDistNoJump = nansum(vectorDistanceList(firstPtIdx:point1_idx-1));
   vorDist = vorDistNoJump + nansum(vectorJumpDistanceList(firstPtIdx:point1_idx-1));
end

%either of these datas may be set to true if the point is near a
%corner; if both are true, the interhatchDist will be set to
%1000000, which should flag it as a bug.
if overwriteForward == true
   forwardDist = 1000000;
end

if overwriteVor  == true
   vorDist = 2000000;
end

if vorDist > forwardDist
   interhatchDist = forwardDist;
else
   interhatchDist = vorDist;
end


ammendList = single([nth_void, interhatchDist, pdist([location(1),location(2);point1(1),point1(2)])]);

dist_list = [dist_list;ammendList];
```

```
    end
end


%remove nominals >400 from in-layer surface
delList = [];
for q = 1:length(dist_list)
    if dist_list(q,3) >400
        delList = [delList q];
    end
end
dist_list(delList,:) = [];
```