LayerLock: Layer-wise Collision-free Multi-Robot Additive Manufacturing Using Topologically Interlocked Space-Filling Shapes

Vinayak Krishnamurthy[a,b], Laxmi Poudel[c], Matthew Ebert[a], Daniel H. Weber[e], Rencheng Wu[f], Wenchao Zhou[f], Ergun Akleman[d,b], Zhenghui Sha[e]

[a] *J. Mike Walker '66 Department of Mechanical Engineering, Texas A&M University, College Station, TX USA*
[b] *Department of Computer Science & Engineering, Texas A&M University, College Station, TX USA*
[c] *Department of Mechanical Engineering, University of Michigan, Ann Arbor, MI USA*
[d] *Department of Visualization, Texas A&M University, College Station, TX USA*
[e] *Walker Department of Mechanical Engineering, University of Texas at Austin, Austin, TX USA*
[f] *Department of Mechanical Engineering, University of Arkansas, Fayetteville, AR USA*

Abstract

We present *LayerLock*, an approach for synchronous multi-robot additive manufacturing (cooperative 3D printing or C3DP). Our approach is based on Delaunay Lofts, a class of topologically interlocked shapes that are generated by stacking layers of Voronoi partitions of a set of moving Voronoi sites based on wallpaper symmetries. Our approach is based on two key insights. First, each layer of a Delaunay Loft is simply a tessellation of convex polygons allowing for easy division of cells for collision-free simultaneous material deposition. Second, the unique transition of Voronoi cells along the layers naturally leads to topological interlocking, thereby providing better energy absorption ability compensating for the loss of structural strength due to segmented printing. In this work, we constrain our current investigation to a two-robot system and and develop the *LayerLock* algorithm consisting of three steps: (1) a distance-based division of the Voronoi cells at each layer of the Delaunay Loft, (2) a moving-front strategy for determining the sequence of cells for each robot, and (3) print path generation based on the cell sequence, which allows synchronous collaboration. We evaluate our algorithm for a range of geometric parameters such as part orientation and cell resolution. We also demonstrate it practically using a two-robot cooperative 3D printing platform.

Keywords: 3D Printing, Cooperative Manufacturing, Path Planning, Voronoi Decomposition, Topological Interlocking

1. Introduction

1.1. Broader Context

Cooperative 3D printing (C3DP) is a novel form of additive manufacturing, where multiple mobile 3D printing robots work together simultaneously to print a large-scale object [1]. Cooperation between robots allows the printing to be carried out in parallel, reducing the overall print time without compromising on the print quality. Additionally, the size of the print object is not limited by the build volume of the printer, allowing mobile printers to print an object much larger than themselves. Thus, C3DP has the potential to scale up both in terms of print speed as well as print size compared to the conventional gantry-based 3D printing method.

C3DP, however, is at a nascent stage and comes with a set of unique challenges such as scalability in size, modular systems design to address trade-offs between resolution and speed, coordination across robots, multi-material support, etc. As a result, many computational approaches (for problems such as part orientation, novel support structures, optimal slicing, etc.) that are currently considered standard in single-robot printing are not directly applicable to C3DP. For example, standard methods for slicing a part volume may fail when considering how multiple robots coordinate, how the part is positioned with respect to the robots, and the interfacial strength between the sub-volumes printed by different robots. Therefore, there is a need for a variety of principled, systematic, and controlled investigations for overcoming these challenges in order to enable the production of arbitrarily complex geometries in the future.

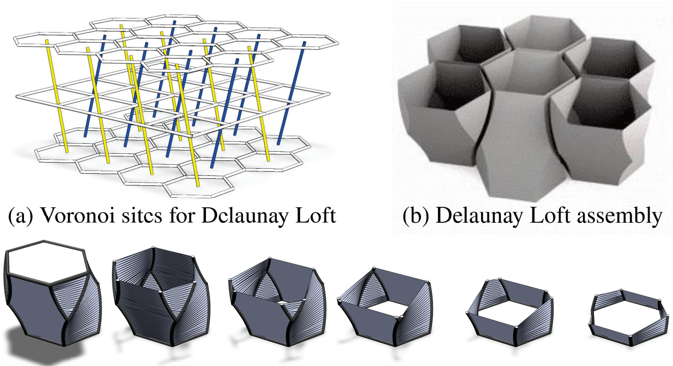
1.2. Problem & Background

While the use of multiple 3D printers to print a part reduces the overall print time, it also requires that the entire part be segmented into multiple sub-volumes [2, 3]. Such segmentation inherently leads to the loss of structural strength of the printed part. Our goal in this paper is to focus on the fundamental problem of interfacial bonding strength between sub-volumes cooperatively printed by multiple robots in a *synchronous* fashion. There are two inter-related components to this problem: (1) design of interface geometry and (2) multi-robot scheduling and path planning.

1.2.1. Design of Interface Geometry

One of the common approaches for part segmentation in C3DP is *chunk-based printing* where a large part is partitioned into smaller chunks (sub-volumes of a part), which are then assigned to the printing robots for parallel printing. In most current approaches these chunks admit sloped but planar interfaces [2, 3, 4]

Email addresses: vinayak@tamu.edu (Vinayak Krishnamurthy), lpoudel@umich.edu (Laxmi Poudel), matt_ebert@tamu.edu (Matthew Ebert), dheinzw@gmail.com (Daniel H. Weber), rw062@uark.edu (Rencheng Wu), zhouw@uark.edu (Wenchao Zhou), ergun.akleman@gmail.com (Ergun Akleman), zsha@austin.utexas.edu (Zhenghui Sha)



(a) Voronoi sites for Delaunay Loft (b) Delaunay Loft assembly

(c) Cell-transitive 2-honeycomb structure of Delaunay Loft

Figure 1: An example of Cell-Transitive 2-Honeycombs: a Delaunay Loft that interpolates hexagonal and square grids. Each layer partitions using the points obtained by line and layer intersections as Voronoi sites. The decomposition is strictly 2D. This guarantees each Voronoi cell is a convex polygon. To guarantee to obtain a Cell-Transitive 2-Honeycomb in each layer, the lines must be closed under one of the wallpaper symmetries. As a consequence, the intersects each layer only once.

that are simple to print but are prone to failure under transversal loading conditions. Recently, Manoharan et al. [5] introduced a corrugated partition for intersected sandwich layers (CPISL) which allows the interaction between two adjacent chunks or sub-volumes in corrugated fashion toward a better *interfacing* between the sub-volumes. However, such an approach results in non-smooth interfaces that may induce high stresses at sharp corners throughout the printed part.

1.2.2. Multi-robot Scheduling and Path Planning

In order to understand the issue of robot scheduling, let us re-examine the choices made by prior work in terms of the geometry of the chunks. Most chunk-based approaches [2, 3, 4] exploit the planar interface because it simplifies robot task scheduling **at the interface** from the perspective of collision avoidance without the need for complex communication protocols [6]. For instance, in case of the approach presented by Poudel et al. [3], one robot is simply able to print on a sloped interface printed by another in a sequential manner.

In contrast to this, approaches such as CPISL by Manoharan et al. [5], require optimal control strategies for collision avoidance specifically to handle the vertically interlocking geometry at the interface of two sub-volumes.

1.2.3. Scope of the Problem

Given a volume to be printed, the problem we focus on in this research is to partition the volume in such a way that: (1) the **interfaces between the sub-volumes are smooth and stronger** compared to planar interfaces and (2) the robots printing this volume work in parallel in a **collision-free** manner **without the need for asynchronous or distributed control**. To address this problem, we develop a novel approach that utilizes the concept of space-filling topological interlocking shapes that are printed by multiple robots in a layer-by-layer manner.

1.3. Approach & Rationale

Our work is inspired by the notion of topological interlocking wherein the core idea is to arrange elements or blocks of special shapes in such a way that the whole structure can be held together by global peripheral constraints, whereas locally they stay in place because of kinetic constraints as a byproduct of their shape and arrangement [7, 8]. As a result of such interlocking, the elements do not require a key or connectors to be held together, eliminating any need for high-precision machining, which can result in stress concentration. In addition to such topological interlocking principle resulting in segmented parts with no significant stress concentrations, it has been experimentally demonstrated that such topologically interlocked assemblies possess favorable mechanical strength, damage tolerance, fracture resistance or toughness, resistance to catastrophic crack propagation, and energy absorption under impact conditions [9, 10].

Most of the topological interlocking assemblies are not space-filling, i.e., there are gaps and voids between the individual elements of the topologically interlocking assemblies. However, 3D printing most frequently requires space-filling properties. Thus, to address this requirement, we leverage a recently introduced space-filling topological interlocking structure that uses the geometrical concept of *Voronoi diagrams or tessellation* [11, 12].

Our approach is based on a specific type of geometry, called Delaunay Lofts, recently introduced in [11]. Delaunay Lofts are a class of topologically interlocked shapes that are generated by stacking the layers of Voronoi partitions of a set of moving Voronoi sites based on wallpaper symmetries. Although the space filling nature of Delaunay Loft was identified in [11], the topological interlocking feature was later realized by Estrin et al. [13]. Our approach is based on two key insights. First, each layer of a Delaunay Loft is a cell-transitive 2-Honeycomb (i.e., a tessellation of congruent shapes) composed of convex polygons. This allows us to allocate polygonal cells among multiple printers for collision-free parallel printing using simple metrics like Euclidean distance. Second, the unique transition of Voronoi cells along the layers naturally leads to topological interlocking, thereby providing better energy absorption ability compensating for the potential loss of structural strength due to segmented printing. Thus, with these insights, we develop the *LayerLock* algorithm to achieve collision-free multi-robot printing. Our algorithm is based on a simple cell labeling strategy in a layer-wise manner. We first label the cells of the 2-Honeycomb in a given layer. We then employ a novel "*moving fronts*" strategy to determine the sequence of cells that will be printed by individual robots. Finally, we generate the print path along the previously determined cell sequence for each robot.

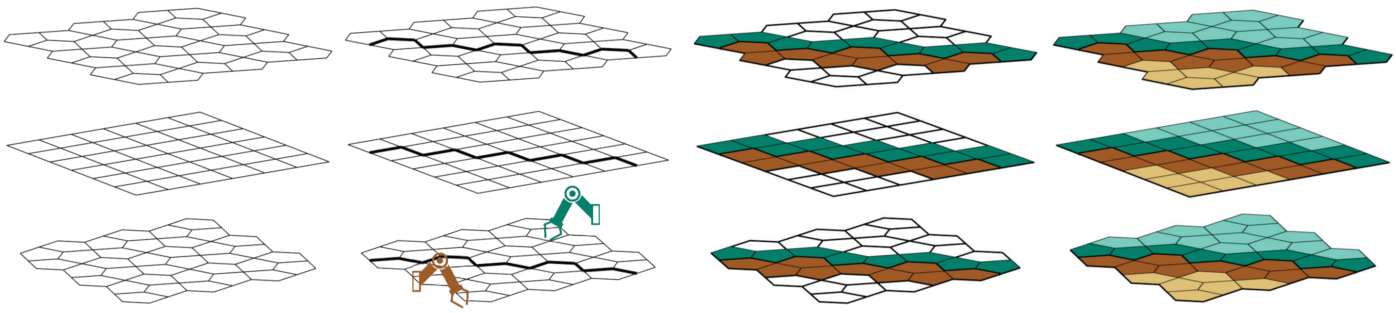
1.4. Key Contributions

We make three main contributions in this work.

1. We introduce a new geometry based synchronous strategy for enabling collision-free C3DP. This strategy is based on the following three principles:

   (a) Each robot prints as equal as possible volume of material in a layer-by-layer manner. An equal division of labor minimizes the total time to print a given cellular structure.

   (b) The geometry of the layers is defined such that each layer is a cell-transitive 2-Honeycomb with convex cells. The convexity and congruence of the cells



(a) Layer-wise cell structure (b) Interfacing path based on robot positions (c) Interfacing cell computation (d) Non-interfacing cell computation

Figure 2: Flowchart of the print process (a) We represent multiple layers of a Delaunay Loft. In the example shown here, the base layer is a grid comprised of regular pentagonal cells, mid-layer composed of regular quadrilateral cells and top most layer identical to base layer (b) The cells are assigned to the two robots using the Euclidean distance of the cell centroid from the robot position. Robot base is assumed to be the robot position for the sake of simplicity of calculations. The black path indicates the interfacing path between the cells assigned to the robots (c) Brown and Green colored cells indicates path for interfacing cells. (d) In addition to Brown and Green cells ,Brown cells interfacing with Light Brown cells and Green cells interface with Light Green cells

in each layer enables a simple division of the print-volume without the need for careful path planning for collision-free synchronous printing.

(c) The integration of layers results in volumetric segmentation such that the interfacing between neighboring segments results in either a complete or partial topological interlocking. Not only does this allow for stronger adhesion between neighboring segments but also offers some mechanical advantages such as higher impact resistance and energy absorption in lieu of the structural strength lost due to segmentation [7, 13].

As such, this strategy is generalizable in the sense that it can be easily extended to any type of interlocking structure composed of space-filling shapes [12, 14].

2. To operationalize this strategy we leverage a recently developed class of space-filling shapes known as Delaunay Lofts as an example (Figure 1) and develop a complete algorithmic pipeline that we call *Layerlock*. Our algorithmic pipeline utilizes a simple distance based cell labeling scheme to divide a convex 2-Honeycomb layer between two robots along an interfacing path (Figure 2(b)). Subsequently, it leverages the topology of the layer tessellation to determine a collision-free sequence of cells (Figure 2(c)) while maintaining simultaneity of the printing process between the two robots (Figure 2(d)).

3. We conduct a series of numerical experiments to characterize the effects of different geometric parameters such as the resolution of the Delaunay Loft cellular structure and the angle of the printed volume with respect to the two robots printing the volume. Our analysis confirms that an as equal as possible distribution of the volume on a layer-by-layer basis indeed minimizes the total printing time for a given volume.

4. We demonstrate the *LayerLock* algorithm in action by printing four example Delaunay Loft structures using two SCARA arm robots equipped with print-heads in collision-free manner. The visual inspection demonstrates strong adhesion between the segments of the volume printed by two separate printing robots.

2. Literature Review

As an emerging new approach to additive manufacturing [15], there are few seminal works related to C3DP in the existing literature. This is particularly true for the C3DP of the topological interlocking structures. In the following sections, we review the relevant studies on five subtopics: 1) existing approaches to C3DP, 2) slicing for multi-robot AM, 3) cellular structures in 3D printing, 4) interlocking structures in AM and 5) topological interlocking in AM.

2.1. Existing Approaches to Cooperative 3D Printing

Zhang et al. presented a large-scale concrete structure that was printed using two mobile robots with SLAM technology (simultaneous localization and mapping) [16]. The study presents a pipeline for printing using multiple robots consisting of multi-robot placement optimization, platform navigation and localization, nozzle trajectory planning and execution. However, the parts printed by the cooperating robots lack any interlocking features. Shen et al. demonstrated the use of four large industrial robots for printing [17]. Such collaboration was achieved by dividing the work space into several safe and interference areas and developing scheduling algorithms based on efficiency egalitarianism for conflict-free printing. Similarly, Manoharan et al. presented a work that segmented a part into two equal sub-volumes to be allocated to two printing robots [5]. To enhance bonding issues between the two volumes, they presented a corrugated partition methodology for the intersected sandwich layers, which allows individual robots to print an alternate layer in the boundary region. The simulation results show promise; however, the physical implementation requires constant monitoring and communication between multiple components of the system, which can be unreliable. Thus, it is not clear how the system would perform in real printing scenarios.

2.2. Slicing for Multi-Robot Additive Manufacturing

Although the conventional slicing algorithms are based solely on the geometry of a part, this is evolving as researchers realize the need for additional considerations such as integration of the collaboration strategy in slicing, topological features, etc. Djuric et al. presented a modular approach to slicing for multiple agents [18]. They planned to combine part geometry with knowledge-based data to develop a travel path for collaborative robotic printing. However, it is still in the proof-of-concept phase. Similarly, Mao et al. presented an adapted slicing framework that optimizes a slicing plan using dynamic programming based on the input CAD model [19]. In separate studies, the authors of [20, 21] incorporated feature-based design modularity into the traditional layer-by-layer strategy. To enable concurrent printing between multiple extruders, Jin et al. [22] developed a general tool-path allocation and scheduling approach to assign sub-paths at each layer to multiple extruders. These sub-paths are formed by "breaks" in the printing paths that are inherently created by slicing software for single-extruder machines. The authors demonstrated their approach in simulations with three extruders and showed that the layer printing times were reduced by as much as 60% compared to single-extruder machines.

While most of the slicing approaches are focused on planar slicing, Etienne et al. a curved slicing approach demonstrated its use by fabricating many models using a 3-axis printer [23]. Lui et al. proposed an efficient approach to large-scale modeling and slicing of lattice structures, which consumes less memory than conventional triangular meshes [24]. An extended review of planar and non-planar slicing algorithms is presented in by Nayyeri et al. [25]. These approaches to slicing and modeling provide advantage over the conventional planar slicing solely based on geometric features of a part and could be potent tools multi-robot cooperative 3D printing.

While our layer-wise approach is reminiscent of the general principle of slicing, we seek to address a completely different problem than a typical slicing algorithm. Given an arbitrary part geometry, a slicing algorithm primarily deals with how to "*rasterize*" the geometry along the vertical dimension of the print nozzle. In contrast, our goal is to address the lack of principled strategies for cooperative printing such that (a) the inter-printer collisions can be avoided and (b) the interfaces between the sub-volumes printed by two different printers are strong with increased bonding area between the sub-volumes. We posit that our approach, therefore, can complement current or future slicing techniques that are specifically tailored to C3DP.

2.3. Cellular Structures in 3D Printing

The use of cellular structures such as honeycomb and lattices has been popular because they provide superior energy absorption upon impact [26, 27, 28]. Kucewicz et al. studied the crash-worthiness properties of three different types of 3D printed cellular structures to assess the influence of mesh type and mesh size [28]. They concluded that the honeycomb cellular structures had the best energy-absorption and crash-worthiness properties. Similarly, Lui et al. presented a strategy for the design of cellular materials based on the Voronoi-Monte Carlo approach [29]. They demonstrated the ability of the presented approach to vary the density and spatial distribution of cells, which could be used to tune the bending performance of the part using a 3D printed wrench. While these works offer interesting possibilities for achieving advantageous mechanical properties, methods to print such intricate geometries is currently a far-fetched goal from the perspective of C3DP due to operational challenges. Furthermore, such structures are not particularly targeted toward improving the interface between portions printed by different robots. In this regard, an interesting outcome of our approach is that it combines the idea of cellular structures with the idea of interlocked geometries that is specifically tailored for C3DP.

2.4. Interlocking Structures in Additive Manufacturing

Interlocking, as a concept, is particularly intriguing for C3DP especially because it offers the promise of allowing better interfacing between portions printed with different robots. One of the early uses of interlocking can be found in the work by Song et al. where a voxelization-based approach was presented to partition a large part into small volumes that can fit within the build volume of the printer and can be connected to each other by 3D interlocking [30]. This, however, is an example where the part is obtained through the assembly interlocking once each interlocking piece has been printed individually. Our goal is to embed such interlocking as an integral part of the printing processes itself.

A few previous works have effectively utilized interlocking in additive manufacturing, especially for multi-material printing where the interfacial bond-strength is critical for maintaining part integrity [31]. Beecroft explored the use of Selective Laser Sintering (SLS) printing using nylon powder to create a flexible topological interlocking structure (weft knitted structures) [32]. Similarly, Malik et al. presented a design and optimization of sutured material with jigsaw-like geometry to predict the relationship of the interlocking angles of the jigsaw with pullout strength and energy absorption and concluded that both the pull-out strength and the energy absorption increase with higher interlock angles [33]. This idea was further reinforced by Ribeiro et al.[34] wherein they investigated U-, T-, and dove-tail geometries to investigate interlocking for multi-material printing. An important issue with these ideas is that the jigsaw-like interlocking works well mainly for tensile loads applied in the plane containing the locking geometry. Because of these desirable properties, the interest in leveraging interlocking as a principle for multi-material additive manufacturing has increased. For instance, Dijkshoorn et al. [35] demonstrated an interesting interlocking-based approach to enable electrical wire connections in 3D printed circuit components. Another recent work by Mustafa and Kwon [36] developed an approach where a infills for parts to be produced with different materials are merged through geometric intertwining to improve interfacial bond-strength.

2.5. Topological Interlocking in Additive Manufacturing

In our work, we introduce a methodology that leverages the principle of topological interlocking as a means to increase interfacial bonding between portions of a given part printed by two different printers. A unique characteristic of a topologically interlocking assembly is that the motion of each interlocking piece is restricted along the normal to the plane where peripheral forces are acting [37, 38]. As a result, topologically interlocking structures have been shown to have interesting mechanical properties such as high impact absorption and high fracture toughness [10, 9, 39, 7, 40].

In contrast to previous jigsaw-type geometries, Duty et al. demonstrated Z-pinning method (inserting a pin in Z-direction to create interlocking properties) to decrease the anisotropy and increase the mechanical strength and toughness in Z-direction (direction of printing) [41]. Their work is reminiscent of weave-like

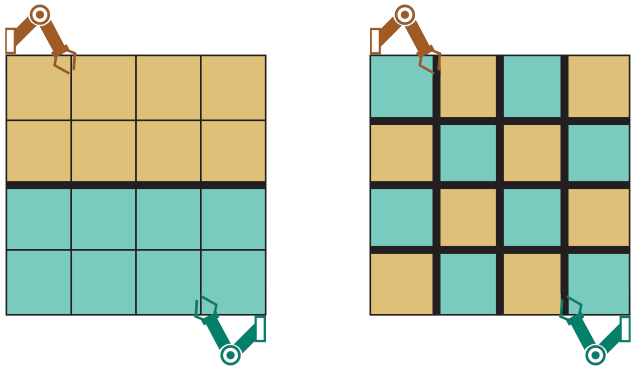


Figure 3: Voronoi decomposition of two robot positions over a regular grid gives a simple boundary that can be used as interfacing path (left). On the other hand, a random assignment such as a checkerboard gird pattern generates a complex boundary with multiple loops, which results in an impossible interfacing path.

structures that interlace amongst each other to achieve these improvements. More recently, Kuipers et al. advanced this idea further for multi-material printing by introducing interlaced topologically interlocking lattice (ITIL) [42]. They adopt a different definition of topological interlocking wherein an interlocking is considered to be topological on the basis of invariance under continuous deformation. While these are seminal works, they are both tailored for continuous extrusion either one or multiple materials are being interlaced sequentially by a single printer. Therefore, applying these strategies in multi-robot printing will require complex distributed and asynchronous control to achieve collision avoidance and optimal printing time simultaneously.

Invoking the original definition of topological interlocking, our work leverages a special sub-class of shapes [11, 12] that are space-filling and simultaneously topologically interlocking in such a way that they naturally lend themselves to **collision-free layer-wise synchronous printing by more than one robot**.

3. Conceptual Framework

The goal of our work is to develop an algorithm that allows for collision-free simultaneous printing of a given volume. In other words, collision avoidance and optimal print-time are both equally important in our approach. In our work, we assume a two-robot system that is cooperatively printing a single shape bearing in mind that the underlying principles behind this framework are generalizable to more than two robots. We further assume that the robots are "*symmetrically located*" with respect to the volume being printed. We note that even with two robots under these assumptions, the problem of motion planning is quite involved [43]. Therefore, we currently constrain our investigation to two robots for practical simplicity and we model each robot as a point identified as its fixed base frame. Our main contribution in this paper to devise a simple strategy allowing us to develop an effective algorithm that does not need careful motion planning.

3.1. Rationale

In order to address the need for algorithmic simplicity, collision avoidance, and simultaneous print-time optimality, we make use of two fundamental principles. First, each robot should print equal area in each layer, thereby printing equal volume as the second robot. Second, for a given layer, the boundary separating the regions assigned to the two robots should also split the layer in two contiguous regions (i.e. there should be no disconnected components belonging to the same robot). The key insight is that partitioning space based on these principles will results in optimal time for completing the print while simultaneously avoiding collisions. If each robot is not given equal area to print then the robot with less area will end up waiting for robot with more area to finish a given layer. To understand the second principle better, consider a square domain that is split into an arrangement of square cells (Figure 3). One can either divide this domain into two equal rectangular domains as split by a single straight line (Figure 3 left panel) or a checkerboard pattern (Figure 3, right panel). It is quite obvious that despite equal division of area, the second alternative is the worst case in terms of the probability of collision simply because every single edge in this case is shared by two cells assigned to two different robots. Keeping these observations in mind, our conceptual framework essentially deals with how to appropriately split the volume to be printed equally between the two robots.

3.2. Layer-wise Inter-Robot Partitioning

To share the workload between the two robots, we first partition each layer into a cell-transitive 2-Honeycomb [44] using Voronoi tessellations of points arranged according to wallpaper symmetries (See Figure 4(a) as an example). These 2-Honeycombs are called cell-transitive (regular or isochoric) [45] since they consist of congruent (i.e., identical) cells. Moreover, these congruent cells in our case are guaranteed to be convex polygons because these polygons are Voronoi cells obtained from points on the plane. It is well known from prior literature [46] that Voronoi tessellations of symmetric point-sets result in cell-transitive 2-Honeycombs.

In order to partition the layer between two robots, each cell is assigned to the robot that is closer to the center of the cell (Figure 4(b)) However, in our case, we end up with a piece-wise linear path along the edges that are shared by cells belonging to different robots. We call this the ***interfacing path*** (Figure 4(c)), which is formally defined as the union of edges and vertices shared between adjacent cells in a grid structure. The interfacing path depends on the cell classification using the two robot positions. Note that *interfacing path changes in each layer* since the shapes of cells changes layer-by-layer, even though the two robots (their base frames) remain static (Figure 2(b), (c)). It is this change that enables better interlocking between the two regions. Furthermore, note that if the 2-Honeycomb is symmetrically places between the robots, equal number of cells will be assigned to each robot resulting in equal area being printed by each robots.

3.3. Collision-free Cell Labeling

The interfacing path acts as a hard boundary between the cells assigned to the two robots. Note that the cells that contain an edge or vertex in the interfacing path are define the regions where collision is possible. In other words, if both robots simultaneously print the cells along the interfacing path, there is a chance of collision. We call these *interfacing cells*. Also note that any cell that is not attached to interfacing path is collision free. We

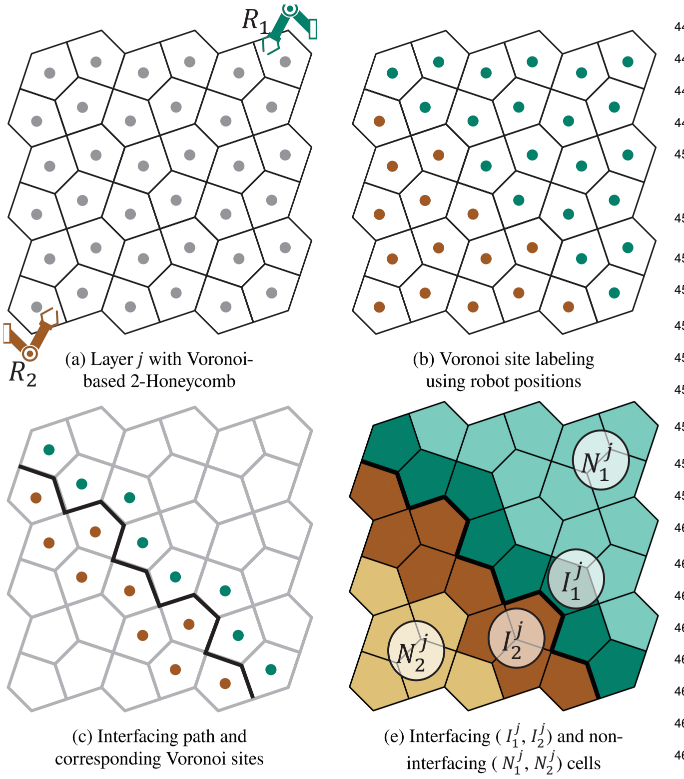


Figure 4: The conceptual framework for *LayerLock* is based on equal partitioning of a given layer. For a given layer j and robot positions R_1 and R_2 (shown in a), we perform a preliminary cell assignment to two robots (shown in b), compute the interfacing path (shown in c), and finally perform collision-free cell labeling (shown in d) to obtain interfacing and non-interfacing cells. This process is repeated for each layer independently.

call these *non-interfacing cells*. Based on this, we can further sub-divide the cells into four regions as follows:

- I_1^i: **Interfacing cells** in layer i printed by robot R_1.
- N_1^i: **Non-interfacing cells** in layer i printed by robot R_1.
- I_2^i: **Interfacing cells** in layer i printed by robot R_2.
- N_2^i: **Non-interfacing cells** in layer i printed by robot R_2.

Each individual cell is labeled based on this classification. The regions I_1 and I_2 are those where two robots can collide. The robot R_2 should not start any cell in the I_2 until robot R_1 completes printing all the cells in the region R_1 and leave to print N_1. On the other hand, the regions N_1 and N_2 are collision free. In other words, these regions can be printed simultaneously. Using this simple scheme, we circumvent the need for complex path planning algorithms that rely on constant inter-robot communication and intermittent waiting.

3.4. Interlocked Layering

In themselves, the cell labeling strategies outlined previously are not enough for producing overall shapes that are interlocked. In order to achieve interlocking we draw from recent works on Delaunay Lofts by Subramanian et al. [11] and generalized Abeille Tiles by Akleman et al. [12]. Interestingly, both these works demonstrate the generation of interlocked structures in a layer-wise manner with each layer being a 2-Honeycomb. However, Delaunay Lofts offers the additional advantage of convexity — each cell in each layer is a convex polygon. Therefore, we chose to adapted the concept of Delaunay Lofts wherein we create our layer-wise 2-Honeycombs based on the 2D Voronoi decomposition of points that are closed under a 2D symmetry group. These points are obtained as an intersection of a control curve with each layer (Figure 2).

3.5. Overview of the LayerLock Algorithm

Based on the decomposition of each layer, the printing algorithm is straightforward. For a given layer L_j we first have robot R_1 print its interfacing cells (I_1^j). Since this region can have collision, robot R_2 waits until R_1 finishes printing the region I_1. Once the print of I_1 is completed, robot R_2 starts to print the cells I_2^j and R_1 starts printing cells in N_1^j. Once R_1 completes printing the region N_1^j, it goes to layer L_{j+1} starting at I_1^{j+1}. Since there is an approximately the same lag between the the robots, when R_2 completes the printing of layer L_j, R_1 almost completes printing the region I_1^{j+1} of the layer L_{j+1}. Let us assume that we print $2n$ layers. Assuming that each layer takes exactly the same time t_L, the total printing time with only one printer using traditional 3D printing is $2nt_L$. With two C3DP robots, we can obtain nt_L at best. Using this process, R_2 only waits idle in the first layer. The total collision area of I_1^j is usually less than 25% of the total area. As a result, the total printing time becomes approximately $(n + 0.25)t_L$. In other words, this straightforward algorithm leads to a near-optimal solution for large n's. This gives only a rough estimate, see Section 5 for actual printing times. In the next section, we provide the details of the *LayerLock* algorithm.

4. *LayerLock* Algorithm

Given a volume to be printed, we first compute a layer-wise partitioning of the volume using point symmetries described in the work by Subramanian et al. [11]. As a result, we get a sequence of layers $L = \{L_j\}$ where each layer L_j is a cell-transitive 2-Honeycomb. Each layer is represented as a planar polygonal mesh, $L_j = (V_j, F_j)$. We assume that all cells F_j are ordered consistently. Further, $\{\mathbf{c}(f)\}$ is a list of centroids of the cells $f \in F_j$.

The *LayerLock* algorithm is a layer-wise algorithm that takes each layer as an input and generates the complete paths P_1 and P_2 for robots R_1 and R_2. The algorithm is comprised of three main steps, namely, (1) computation of the interfacing path and collision-free cell labeling, (2) computation of the cell sequences for the two robots based on moving fronts, and (3) computation of the path based on the cell-sequence. These steps are detailed below for a given layer L_j with the assumption that all computations are planar.

4.1. Step 1: Interfacing Path & Collision-free Cell Labeling

Given a layer $L_j = (V_j, F_j)$ and two robot positions $\mathbf{r}_1$ and $\mathbf{r}_2$, we first separate the cells into two subsets F_1^j and F_2^j such that $\|\mathbf{c}(f) - \mathbf{r}_1\| < \|\mathbf{c}(f) - \mathbf{r}_2\|$ for all cells in F_1^j and the remaining cells belong to F_2^j. We then traverse through each cell in F_1^j and F_2^j and label the edges and vertices as interfacing or non-interfacing.

Consider an edge e that is shared by two cells f and g. The edge is labeled as an interfacing edge if $f \in F_1^j$ and $g \in F_2^j$. Similarly, a vertex v shared by two cells f and g is labeled as an interfacing vertex if $f \in F_1^j$ and $g \in F_2^j$. Subsequently, we add f

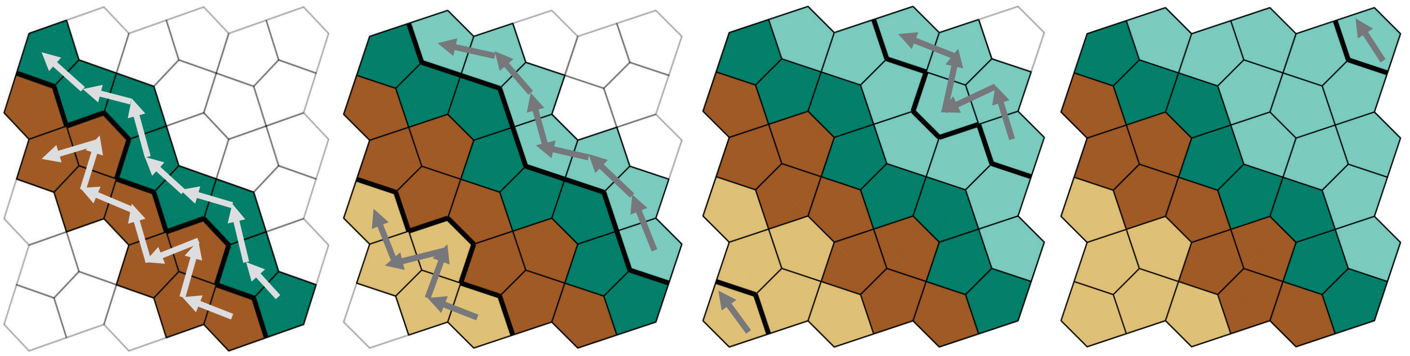
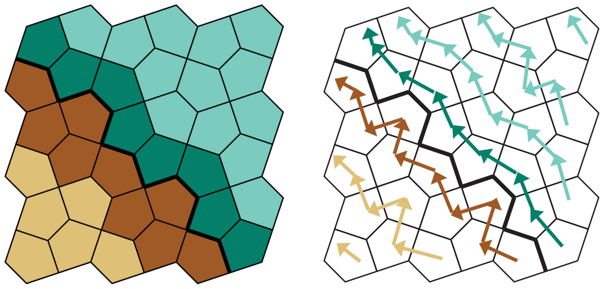


Figure 5: This illustration of cell sequencing is shown. Starting from the interfacing path (left most image), we compute moving fronts in a direction away from the interfacing path toward the robot positions. At each iteration, a subset of cells for each robot is sequenced based on the moving front and the process is repeated until no cells are left non-sequenced (right most image).



(a) Labelled cells (b) Cell sequence

Figure 6: Cell sequencing is illustrated for a pentagonal 2-Honeycomb. Given the locations of the two robots we first compute the interfacing path and label the cells of the honeycomb into four categories, namely, the interfacing cells (Brown and Green in a) and the non-interfacing cells (Light Brown cells and Green cells in a). The interfacing cells are all cells that share a vertex or an edge in the interfacing path. Based on the moving fronts strategy shown in Figure 5, we determine the sequence of cells (shown in c).

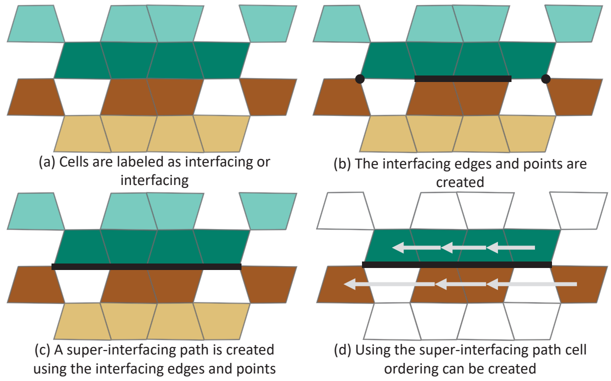


Figure 7: Once cells are labeled as interfacing or non-interfacing (a) the edges and points that lie on the boundary of these cells can be obtained which may be disconnected (b). In this case a super-interfacing line is created by traversing mesh connectivity (c). Using the super-interfacing path the cell sequence is the cell order along the direction of the line (d).

to interfacing cells I_1^j of robot R_1 and g to the interfacing cells I_2^j of robot R_2. After traversing all of the cells in F_1^j any cell that is not in I_1^j is added to N_1^j. The same is done for F_1^j to create N_1^j.

Once all faces are traversed, we obtain a set of edges and vertices that construct the interfacing paths. To construct our path, we simply order the vertex indices of the edges in the sequence, say $\{i_1, \dots, i_q\}$ where the edges are given by $IP = (i_1, i_2), \dots, (i_{q-1}, i_q)$. If the interfacing vertices are not contained within IP then additional computation is needed to form the complete IP (See Section 4.1.1). At the end of this step, we get the interfacing path (IP), the set of interfacing cells for the two robots (I_1^j, I_2^j), and the set of non-interfacing cells for the two robots (N_1^j, N_2^j).

4.1.1. Disconnected interfacing edges and points

In some cases there are interfacing edges and points that cannot form one continuous path (Figure 7(a)). In this case a 'super-interfacing' path is computed which connects all interfacing edges and points. To compute this we consider the vertices which appear on the interfacing edges and points $\{v_1, v_2, \dots, v_{q-1}, v_q\}$. The interfacing path ($IP$) is obtained by connecting all vertices through the cell structure using the least number of edges. With this computation is is possible that non-interfacing cells contain an edge within the interfacing path, in these cases cells do not change their classification as interfacing or non-interfacing.

4.2. Step 2: Cell Sequence Computation

In the second step, our goal is to use the interfacing path and the cell labels to compute the sequence in which each robot will print its respective cells. Our strategy to sequence the cells is based on a simple moving fronts strategy (Figure 5) applied to each robot independently.

Consider robot R_1 and the cells $F_1^j = I_1^j \cup N_1^j$ associated with it. We begin by sequencing the interfacing cells I_1^j. In order to sequence the cells in I_1^j, we traverse the interfacing path IP in a vertex-edge-vertex fashion, i.e. $i_1 \rightarrow (i_1, i_2) \rightarrow i_2 \dots (i_{q-1}, i_q) \rightarrow i_q$. For each vertex (or edge) in this sequence, we find a cell $f \in I_1^j$ that contains the vertex (or the edge). Note that a cell must either contain a vertex as a corner or an edge, not both. As a result, we re-order the cells in I_1^j based on the the path and label them as *sequenced*.

Once done, we compute a *front*, which is defined as an sequence of edges that separates the sequenced cells from the non-sequenced cells. Notice that this front can be computed exactly using the steps to compute the interfacing path. Once computed, we use the front to sequence a sub-set of cells in N_1^j that is currently labelled non-sequenced and contains the vertices and edges of the current front. We repeat this process until all cells in N_1^j are sequenced. The same process is also applied to the cells $F_2^j = I_2^j \cup N_2^j$, which are assigned to the second robot (Figure 6).

4.3. Step 3: Robot Path Computation

Once we have the sequence of cells to be printed, computing the print path is rather simple. As such, there are different types of paths (zig-zag, spiral, etc.). However, given the unique structure of our layers as well as the convex shapes of each cell, we chose to use contour paths. For each cell in the sequence, we simply create the contour paths using the offset curves of the polygonal cell. For each cell, we generate a path from the outermost contour to the inner-most contour. Once a cell is printed, the print-head moves from its current location inside the cell to the closest vertex of the next cell in the sequence.

5. *LayerLock* Numerical Analysis

One of the primary motivations for C3DP is that the simultaneous printing of different portions of a volume result in significant reduction of time. In our case, the total time for printing a part with two robots broadly depends on two factors: (1) the time taken to deposit material and (2) the time taken to move between cells and across layers in a sequence. Note that these factors, in turn, depend on the specific geometry and the resolution (i.e. the number of cells in a given volume) of a Delaunay Loft structure. Therefore, we conducted two systematic numerical studies to investigate the effect of these two factors on the print time.

5.1. Computing the Print Time

Computing the exact printing time is an involved task as the velocity profile of the print-head depends on several factors including the length of a given straight path, the number of corners, etc. In our analysis, we simplify the computation of our print time by assuming a constant *effective* speed of the print-head. Specifically, we assume that the head prints a straight line segment starting from rest, linearly increasing to some maximum speed and then linearly decelerating by the time it reaches the end of the segment. Therefore, for a given maximum speed, we can compute the effective speed S by using the speed-time curve. We also assume that the effective speed is the same regardless whether the printer deposits material or moved between cells.

Recall that the basic premise behind our approach is to minimize print time while simultaneously avoiding collision. The way we achieve this is to have the first printer (say R_1) print the interfacing cells of the first layer (i.e. I_1^0). Once this this done, the remaining process is performed in parallel between the two robots. Note that R_2 will invariably end up printing at the end. With these assumptions and considerations, the total time to print is given by $T = \frac{D_{lead}}{S} + max(\frac{D_1}{S}, \frac{D_2}{S})$. Here, T_{lead}. Here, D_{lead} is the length of the path traversed by R_1 while printing I_1^0, and D_1 and D_2 are the lengths of the paths traversed by R_1 and R_2 respectively for the remainder of the process.

5.2. Test Cases

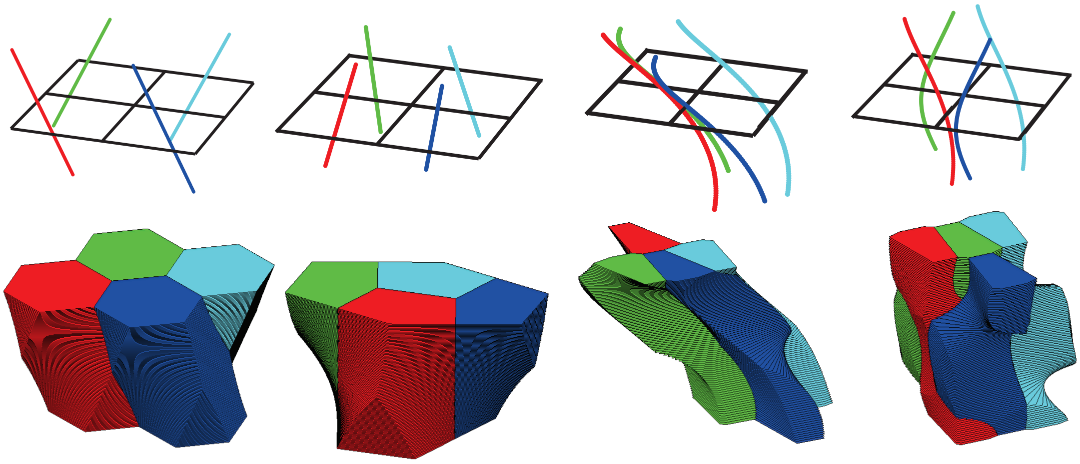
We selected four different varieties of Delaunay Lofts as to conduct our numerical analyses. Our choices was based on the simplicity of the 2-Honeycomb structures in each of the geometries, the overall variety of the structures, most importantly the degree of topological interlocking afforded by each of the structures as listed below:

1. **Hex-Quad-Hex (6-4-6)**: Hex-Quad-Hex starts with a base layer consisting of hexagonal cells. Transition along the control curve yields quad cells in the mid layer and transition back to hexagonal cells in the top most layer.
2. **Pent-Quad-Pent (5-4-5)**: Pent-Quad-pent starts with a base layer consisting of pentagonal cells. Transition along the control curve yields quad cells in the mid layer and transition back to pentagonal cells in the top most layer. Estrin et al. [13] noted that pent-quad-pent shape is bounded by hyperbolic-paraboloid shape in all sides thereby making it completely topologically interlocking.
3. **Helix 1 (6-4-6)**: Helix 1 is created by utilizing two helices as the control curves. These two helices are have different phase start angles and are arranged in a checkerboard configuration (Figure 8 (c)). This method means that the top, middle, and bottom layer may not be a regular hexagon, pentagon or quadrilateral, and are instead a function of the parameters chosen.
4. **Helix 2 (6-4-6)**: Helix 2 is created using the same method as Helix 1 but with different parameters. This caused the structure have to have a rapid change of shape in layers and the structure is very different than Helix 1 (Figure 8 (d))

5.3. Experiment Design

In our analyses, each of the three cases were assumed to occupy a bounding box of dimensions $250mm \times 250mm \times 50mm$ ($L \times W \times H$). we also assume the effective speed to be $40mm/s$. Using these parameters, we perform two simulated experiments to investigate the print time for the three Delaunay Loft test cases.

1. **Rotation Test**: A change in the part's orientation will effectively lead to a change in the interfacing path as well as the collision-free cell labeling. Therefore, we study the effect of the orientation of the print volume with respect to the two robots. Specifically, we assume that the center of mass the structure is coincident with the mid-point of the line joining the two robots. With this configuration, we simply rotate the line joining the robots in increments of 10° (Figure 9). Subsequently, we compare the print times for every angle.
2. **Resolution Test**: Even though changing the resolution of the structure may have little effect on the volume, it may still affect the print time because the number of moves between cells increases with resolution. Therefore, we study the effect of the cell resolution on the print time. For each of the four test cases, we generated structures with a wide range of resolutions (Figure 10 gives an example of a single layer of a pent-quad-pent case). Note that changing the resolution would significantly change the total volume printed because



(a) Hex-Quad-Hex (b) Pent-Quad-Pent (c) Helix 1 (d) Helix 2

Figure 8: All of the cases that were tested are shown with their control curves and the resulting layers shown transparent. A color gradient was applied along the height to show the transition of the shape.

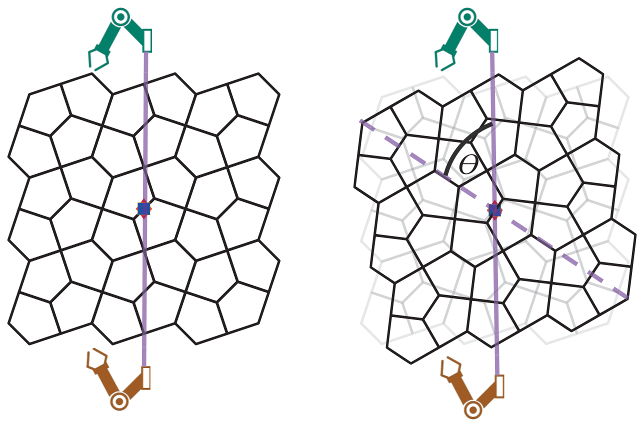
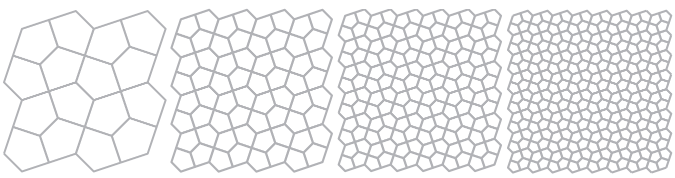


Figure 9: Rotation test : For illustration, we consider base layer of a pent-quad-pent shape.Purple line joining Green and Brown robots is termed as “robot axis”. Centroid of the shape and center of the robot axis are denoted by Blue square and Red diamond shapes respectively. The two centers coincides and lies on the robot axis. Figure(Left) is non-rotated while Figure(Right) represents the shape rotated by an angle θ about the an axis passing through the centroid of the shape .



(a) 4x4 (b) 8x8 (c) 12 x 12 (d) 16 x 16

Figure 10: Resolution test : For illustration purpose, we consider base layer of a pent-quad-pent shape.We represent the same area in each of the square boxes with varying cell numbers.Number of cells (n) (a) 16 (b) 64 (c) 144 (d) 256

of the non-linear shape of the Delaunay Loft. Therefore, we scaled each part such that the parts generated for each resolution occupied the same volume.

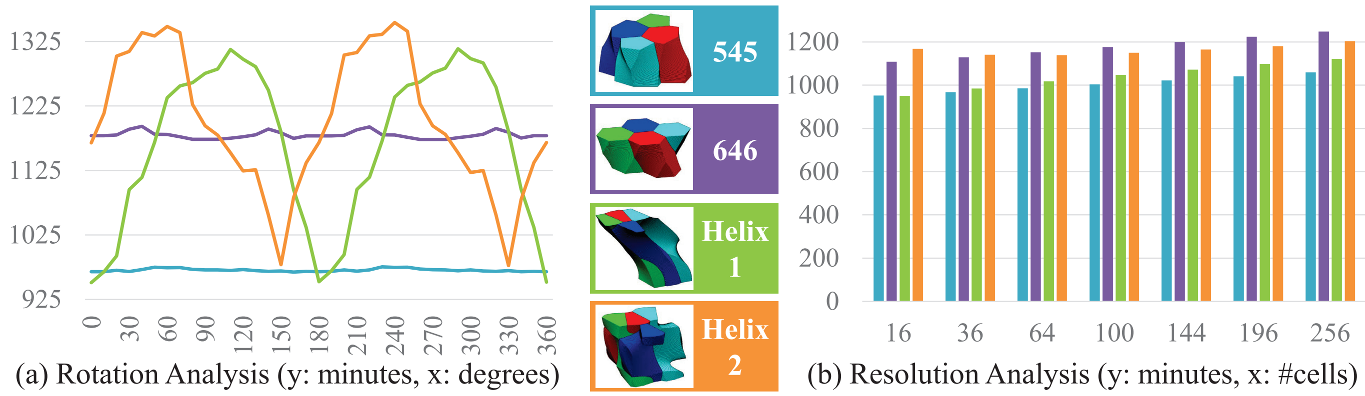
5.4. *Key Findings*

There are three main observations we make in our rotation test. First, each test case exhibited periodic increase and decrease in the time taken. This is expected due to the cyclic nature of angular variation. Second, upon inspection of the data, we found that the increase in time occurs at precisely at angles where the volume division is unequal. This may happen dues to sudden shifting of the interfacing cells from one robot to the other when they are very close to the perpendicular bisector of the line joining the two robots. This is also in line with our initial premise that equal division of labor minimizes print time.

Third, we observe a maximum difference of 8.1 minutes (maximum: 975.5 minutes, minimum: 967.4 minutes) for pent-quad-pent case, 20.4 minutes (maximum: 1193.3 minutes, minimum: 1172.9 minutes) for hex-quad-hex case, 362.6 minutes (maximum: (1313.6 minutes, minimum: 951.03 minutes) for Helix 1, and 377.0 minutes (maximum: (1354.4 minutes, minimum: 977.4 minutes) for Helix 2. Lastly, we note that the pent-quad-pent case took the least average time (970.43 minutes), followed by Helix 1 (1165.1.43 minutes), followed by the hex-quad-hex (1179.8 minutes), followed by the Helix 2 case (1199.8 minutes).

While the hex-quad-hex and pent-quad-pent structures have relatively low change in time regardless of angle the helix structures had large changes in time taken. This is because at certain layers one robot may only have 4 cells to print while the other robot has to print 12 cells. This is the cause of the large difference in printing times based on angle.

In terms of the resolution analysis, we observed a monotonic increase in time taken to print in almost every case. Specifically, we observe a maximum difference of 1.78 hours (maximum: 1059.4 minutes, minimum: 952.3 minutes) for pent-quad-pent case, 2.3 hours (maximum: 1247.7 minutes, minimum: 1108.6 minutes) for hex-quad-hex case, 2.8 hours (maximum: 1121.4 minutes, minimum: 950.0 minutes) for the Helix 1 case, and 1.09 hours (maximum: 1204.1 minutes, minimum: 1138.9 minutes)



(a) Rotation Analysis (y: minutes, x: degrees)   (b) Resolution Analysis (y: minutes, x: #cells)

Figure 11: The results of our rotation test (left) show that the time taken for each case increases and decreases in a periodic fashion. The resolution test (right) shows that the time to print typically increases with an increase in the number of cells, it is possible for a slight reduction in printing time based on non-optimal rotation angle.

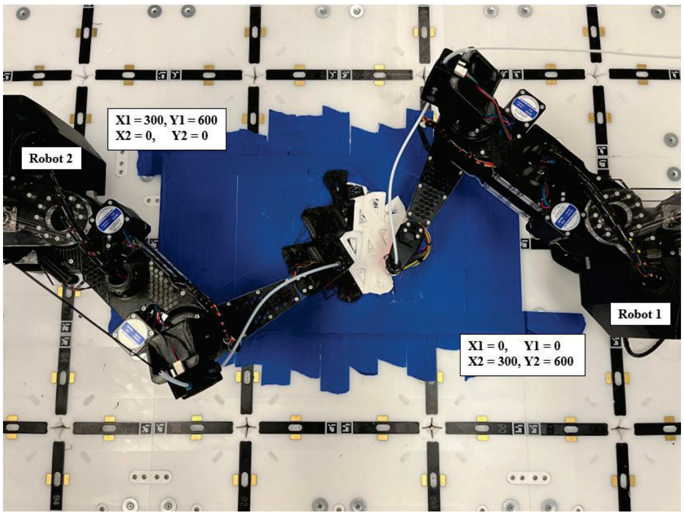


Figure 12: The physical setup of the printing platform for multi-robot cooperative printing using Delaunay loft. X1, Y1 represents the coordinates of robot 1 base and X2, Y2 represents coordinate of robot 2 base.

for the Helix 2 case. An interesting result from the resolution analysis is that for the case of helix 2 there is a decrease of 29.0 minutes when the resolution is increased from 16 to 64. This is because the angle that these resolution analysis were conducted at were 0° which is not an optimal angle for printing time with helix 2. At that angle and low resolution the percentage of cells that one robot prints vs the other robot is very different. As the resolution increases there is a more equal split of cells between the robots leading to a better printing time. In all of the other patterns at 0° rotation the printing time is near optimal. When comparing 16 cells and 100 cells helix 2 has a decrease of 3.5 minutes whereas every other pattern has an increase of more than 1 hour.

Our findings indicated that while it may be straightforward optimize the print time for rotation based on robot labor division, there is an interesting trade-off to consider between resolution (which will dictate the fineness of the structure) and the time taken to print.

6. Physical Results

6.1. Physical setup

The physical experiment was conducted using a platform manufactured by AMBOTS Inc (Figure 12). The setup consists of:

- 2x printhead carrying SCARA robots
- 1x build Plate
- 1x mobile platform
- 16x floor tiles

The printing is carried out by two SCARA printing robots that are mounted onto the floor tiles. The printers have the maximum reach of 350mm. A non-heated build plate, covered in tape for part adhesion, of dimension 300$mm \times$ 600mm, is placed between the two printers prior to printing the input structure. This particular spacing between printers ensures that no collisions occur with the printer elbows, since the printing path generated from the G-code file considers the nozzle as a point rather than a robotic arm. A summary of the technical parameters associated with the SCARA printers is presented in Table 1. While the mobile platform is part of the physical setup, it was not used for carrying out the experiments. More detailed information of the platform and its individual components, including the detailed specifications and working principles, is provided in the work by Poudel et al.[47].

Once a structure is created based on Delaunay Loft using the *LayerLock Algorithm* (Section 4.3) the path sequence is exported in a text file. To undertake the printing task, the path sequence needs to be converted to the G-code instructions. However, the printing robots do not share the same coordinate space, and have their own coordinate space (as shown in Figure 12). This poses a problem because the structures are created with assumptions that the robots have same coordinate space. Thus, to ensure that the two structures align properly at the interfacing path (Figure 6), the structure for one of the robot has to be transformed (rotated and translated). Once the rotation and translation operation is complete, G-code files for printing the individual structure are created and uploaded to the individual robots for printing. The printing parameters used for the printing process are summarized in Table 2.

Table 1: Technical parameters associated with the SCARA printers of the physical setup

Parameters	Values (units)
XY reach	50mm-350mm
Max Z-height	300 mm
Filament Feed	Bowden, 1.75mm
Nozzle	Single extruder
Max. Temperature	295° C
Hot End	Single extruder
X/Y Motion	2-Axis SCARA
Z motion	300mm guide motion driven by lead screw
Min. Layer resolution	10 µm
Max. Print Speed	50 mm/s
Print repeatability	5 µm
Power consumption	20 W
Connectivity	Wireless

Table 2: Printing parameters for the experimental study

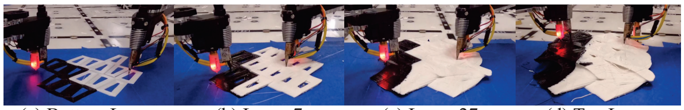
Parameters	Value (units)
Number of robots	2
Nozzle Temperature	240° C
Nozzle Diameter	1.0 mm
Print Speed	20 mm/s
Layer Height	0.40 mm
Line Width	0.45 mm
Material	PLA
Filament Diameter	1.75 mm

6.2. Case Studies

The efficacy of our proposed *LayerLock* algorithm is demonstrated by printing the four shapes described in Section 5.2, using the settings outlined in Table 2, The test cases were printed in a roughly $250mm \times 250mm$ size. The Hex-Quad-Hex and Pent-Quad-Pent were printed with 30 contours, and Helix 1 and Helix 2 were printed with 20 contours to allow the structural testing to highlight the interlocking geometry rather than material properties. As a note, the results in Figure 11 were generated with these contour numbers. All prints had 16 cells in a roughly 4×4 configuration. As shown in Section 5.4, this configuration gave consistently lower print times while still allowing for non-interfacing cells to exist. Using the optimal angle test conducted (Figure 11) a rotation angle of 0° was the optimal or near optimal printing time for 3 of the cases because it most effectively split robot work distribution. For this reason all cases were printed at 0° of rotation. Since Helix 2 has a non-optimal printing time at 0° rotation we also printed an additional version at 150° rotation to confirm this result.

6.3. Printing Results

The printed grid structures have good contact between the adjacent polygons printed by the different robots (Figures 13 and 14). A closer inspection of the printed parts reveals that the bond is consistent throughout the entire boundary. To demonstrate the connection between the cells and the structural strength of the parts, weights were allowed to hang from the printed grids (Figure 15). A comprehensive analysis of relationship between factors, such as print resolution and cell structure, and part strength is needed but is not in the scope of this work.



(a) Bottom Layer (b) Layer 7 (c) Layer 27 (d) Top Layer

Figure 13: Layer-wise printing of Helix 1 is shown using two AMBOTS robots working in collaboration. The unequal distribution of labor is also shown with the left robot printing only 4 cells on layer 7 (b) and 6 cells in layer 1 (a).

(a) Pent-Quad-Pent (b) Hex-Quad-Hex (c) Helix 1 (d) Helix 2 0° rotated (e) Helix 2 150° rotated

Figure 14: The final produces are shown for a two-robot printing of the Hex-Quad-Hex, Pent-Quad-Pent, Helix 1, and two rotations of Helix 2 Delaunay Lofts. The white and black color signify the volume printed by each individual robot.

Print time results for the five test cases are shown in Table 3. These results show that the print time predictions are reasonably accurate and highlight the impact of selecting the optimal part rotation. Discrepancies between predicted and actual time are likely due to travel time at the beginning of the part and whenever the robot returns to home during a pause, such as when waiting for the other robot to finish printing a layer. The robot waiting at a specified home location away from the printed part is to prevent dripping of filament and ensure that even in cases where one robot only has a few interfacing cells and no non-interfacing cells there are no collisions between the robots.

Table 3: Printing time results of the experimental study

Case	Time (minutes)
Hex-Quad-Hex	1180.2
Pent-Quad-Pent	1093.8
Helix 1	1179.7
Helix 2 (0°)	1020.0
Helix 2 (150°)	870.4

7. Discussion

7.1. Generalizability of the Approach

The main advantage of our method is that it is based on topological principles making it suitable for future extensions. There are two main components of our method: (1) the division of cells in a given tessellation across multiple robots and (2) the sequencing of the grid cells for each robot based on the moving fronts. In our current implementation of the algorithm, we make two assumptions as elaborated below.

First, we assume that each layer is tessellated in a cell transitive manner (i.e. each polygon is congruent). This is a natural outcome of utilizing the principle of Delaunay Lofts [11]. Our

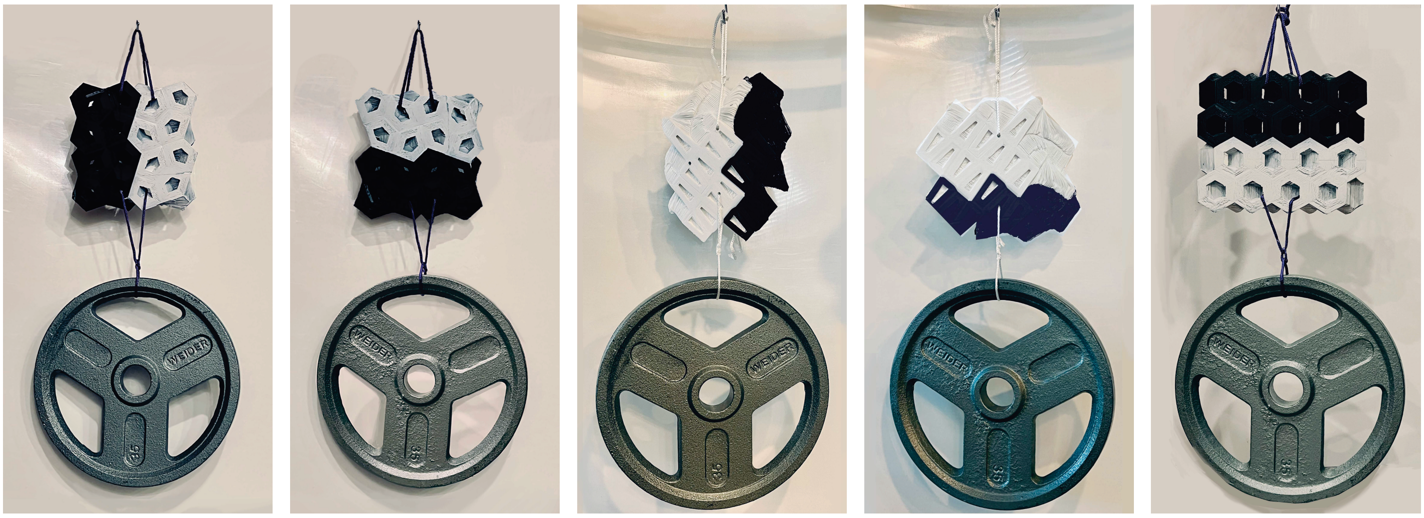


Figure 15: A demonstration of the strength of parts printed with the *LayerLock* algorithm. Parts shown are supported 35 lbs of weight in both the direction perpendicular to interfacing line and parallel to the interfacing line.

use of Delaunay Lofts was specifically to utilize regular partitioning for simplifying as-equal-as-possible division of print volume for each robot in turn enabling near-minimal print time. However, this is a choice that can be removed without any changes to the algorithm. Therefore, in principle, the method works identically for any arbitrary tessellation and even any number of robots as long as the partition for each robot is a tessellation that can be represented as a single connected graph. For instance, our algorithm can be readily applied to other space-filling topologically interlocking shapes such as the generalized Abeille tiles [12] or even more complex interlocking structures such as the bi-axial woven tiles [14].

In keeping with the points above, the second assumption we make is that each layer is a contiguous genus-0 region (i.e. a singly-connected region homeomorphic to a disk). The main implication of this assumption is that the algorithm, in its current form, needs further extension for layers with multiple components — a scenario that would be common for complex shapes. However, we believe that this issue can be resolved by constructing auxiliary topological structures. As a simple example, given a set of disconnected boundaries, one can construct such topological structures using known methods such as Delaunay triangulation on a boundary points, or simply through some heuristic variations on nearest-neighbor graphs. In either case, once the topology is established, our method can be identically applied for cell sequencing and path planning.

In any case, our claim is that our method is general up to the generation of well-bonded interfaces and can be used as one of the elemental steps in future C3DP processes such as slicing, task scheduling, and robot path planning. However, we note that issues such as optimal part orientation, inter-robot communication protocols, and many others will need to be further investigated systematically to achieve this. In fact, partitioning shapes of arbitrary complexity with interlocking interfaces is, in itself, another open-ended problem that needs in-depth investigation.

7.2. *Robot Coordination*

While the final printed grid has good interlock between the half-grids, the physical implementation of the generated algorithms requires solving many key challenges. One of the major challenges in the implementation of collision-free printing is the lack of deterministic certainty when executing the print, resulting in inconsistencies between the theoretical results and the actual printing results. No two printers are identical due to manufacturing tolerances. In addition to this, the execution of the print among different printers is not guaranteed to be synchronized due to run-time uncertainties as well as differences in print distribution per layer to each robot as discussed above. These differences and accumulations of error due to geometric tolerances and uncertainties can result in deviation from the planned printing and print failure due to collision between the two print-heads. In order to ensure a collision-free cooperative printing with multiple robots, additional factors need to taken into consideration while generating the G-code for printing. For arbitrary geometry, additional communication protocols are needed for forced synchronization between printers to avoid collisions caused by the accumulation of differences between printers.

7.3. *Integration for Part Quality & Standardization*

While our algorithm is scalable in a geometric sense, there are other issues such as warping due to thermal contraction of the part, and unsupported overhangs, where extruded material will droop if not layered on top of a support. As such, there are a multitude of methods in materials and 3D printing literature (see [48] for a comprehensive review) that address issues regarding part quality in 3D printing. We believe that our method can be integrated effectively with several existing methods for tackling warping, optimizing for overhangs, etc. For instance, some of the issues that we faced may be addressed in part by printing on a heated bed plate or in a heated chamber and employing a support structure or trimming unsupported edges in peripheral cells.

Given that we implemented our method from the ground up, we did not develop a full-fledged slicer and optimize for the printing process through the control of acceleration and deceleration, retractions, Z-hop, and other parameters. Typically, these issues are handled if the G-code are generated by an established slicer single-robot printing. Overall, we believe our method can be potentially integrated into existing slicers or be further extended into a fully functional slicer for improved part quality.

8. Conclusion

In this paper, we presented *LayerLock*, an effective algorithm to enable collision-free multi-robot cooperative 3D printing. The main idea behind the algorithm was to partition a given volume in a way that the problem of collision-free path planning can be simplified using a geometric approach. To this end, our method leveraged a layer-wise cooperation strategy using cell-transitive 2-Honeycombs that, when stacked, produced interlocking structures. To our knowledge, this is the first approach to devise, analyze, and demonstrate this unique combination of geometric reasoning with multi-robot additive manufacturing. We implemented our algorithm with a large scale two-robot system and demonstrated its efficacy.

While the results are promising to begin with, we believe that we have barely scratched the surface of this problem. For instance, we restricted our investigation to a specific path planning template with only two static robots for simplicity. Secondly, we did not employ any advanced communication protocols to synchronize the two robots. However, we believe that this work offers future directions in terms of partitioning and task scheduling for complex large-scale parts, standardization of robot communication protocols, and extension to advanced robotic systems with mobile robots.

Acknowledgment

This work was partially supported by the National Science Foundation (NSF) Awards #2048182 (Engineering Design and Systems Engineering Program) and #2112009 (Small Business Innovation Research Program). The authors would like to thank the reviewers for valuable comments and feedback. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the NSF.

References

[1] L. Poudel, L. Marques, R. Williams, Z. Hyden, P. Guerra, O. Fowler, Z. Sha, W. Zhou, Towards swarm manufacturing: Architecting a cooperative 3d printing system, Journal of Manufacturing Science and Engineering (2022) 1–11.

[2] J. McPherson, W. Zhou, A chunk-based slicer for cooperative 3d printing, Rapid Prototyping Journal (2018).

[3] L. Poudel, Z. Sha, W. Zhou, Mechanical strength of chunk-based printed parts for cooperative 3d printing, Procedia Manufacturing 26 (2018) 962–972.

[4] L. G. Marques, R. A. Williams, W. Zhou, A mobile 3d printer for cooperative 3d printing, in: Proceeding of the 28th international solid freeform fabrication symposium, 2017, pp. 1645–1660.

[5] M. Manoharan, A. N. Shridhar, V. Y. Vinod, S. Kumaraguru, A novel volume decomposition methodology for multi-robots collaborative additive manufacturing, in: 2020 IEEE 4th Conference on Information & Communication Technology (CICT), IEEE, 2020, pp. 1–6.

[6] S. Elagandula, L. Poudel, Z. Sha, W. Zhou, Multi-robot path planning for cooperative 3d printing, in: International Manufacturing Science and Engineering Conference, Vol. 84256, American Society of Mechanical Engineers, 2020, p. V001T01A034.

[7] Y. Estrin, A. V. Dyskin, E. Pasternak, Topological interlocking as a material design concept, Materials Science and Engineering: C 31 (6) (2011) 1189–1194.

[8] A. Dyskin, Y. Estrin, E. Pasternak, Topological interlocking materials, in: Architectured Materials in Nature and Engineering, Springer, 2019, pp. 23–49.

[9] S. Khandelwal, T. Siegmund, R. Cipra, J. Bolton, Transverse loading of cellular topologically interlocked materials, International Journal of Solids and Structures 49 (18) (2012) 2394–2403.

[10] M. Carlesso, A. Molotnikov, T. Krause, K. Tushtev, S. Kroll, K. Rezwan, Y. Estrin, Enhancement of sound absorption properties using topologically interlocked elements, Scripta Materialia 66 (7) (2012) 483–486.

[11] S. G. Subramanian, M. Eng, V. R. Krishnamurthy, E. Akleman, Delaunay lofts: A biologically inspired approach for modeling space filling modular structures, Computers & Graphics 82 (2019) 73–83.

[12] E. Akleman, V. R. Krishnamurthy, C.-A. Fu, S. G. Subramanian, M. Ebert, M. Eng, C. Starrett, H. Panchal, Generalized abeille tiles: Topologically interlocked space-filling shapes generated based on fabric symmetries, Computers & Graphics 89 (2020) 156–166.

[13] Y. Estrin, V. R. Krishnamurthy, E. Akleman, Design of architectured materials based on topological and geometrical interlocking, Journal of Materials Research and Technology (2021).

[14] V. R. Krishnamurthy, E. Akleman, S. G. Subramanian, M. Ebert, J. Cui, C.-a. Fu, C. Starrett, Geometrically interlocking space-filling tiling based on fabric weaves, IEEE Transactions on Visualization & Computer Graphics (01) (2021) 1–1.

[15] W. Gao, Y. Zhang, D. Ramanujan, K. Ramani, Y. Chen, C. B. Williams, C. C. Wang, Y. C. Shin, S. Zhang, P. D. Zavattieri, The status, challenges, and future of additive manufacturing in engineering, Computer-Aided Design 69 (2015) 65–89.

[16] X. Zhang, M. Li, J. H. Lim, Y. Weng, Y. W. D. Tay, H. Pham, Q.-C. Pham, Large-scale 3d printing by a team of mobile robots, Automation in Construction 95 (2018) 98–106. doi:https://doi.org/10.1016/j.autcon.2018.08.004.
URL https://www.sciencedirect.com/science/article/pii/S0926580518304011

[17] H. Shen, L. Pan, J. Qian, Research on large-scale additive manufacturing based on multi-robot collaboration technology, Additive Manufacturing 30 (2019) 100906. doi:https://doi.org/10.1016/j.addma.2019.100906.
URL https://www.sciencedirect.com/science/article/pii/S2214860419308243

[18] A. Djuric, R. J. Urbanic, Using collaborative robots to assist with travel path development for material deposition based additive manufacturing processes, Computer-Aided Design and Applications 15 (4) (2018) 542–555.

[19] H. Mao, T.-H. Kwok, Y. Chen, C. C. Wang, Adaptive slicing based on efficient profile analysis, Computer-Aided Design 107 (2019) 89–101. doi:https://doi.org/10.1016/j.cad.2018.09.006.
URL https://www.sciencedirect.com/science/article/pii/S0010448518301763

[20] F. Michel, H. Lockett, J. Ding, F. Martina, G. Marinelli, S. Williams, A modular path planning solution for wire + arc additive manufacturing, Robotics and Computer-Integrated Manufacturing 60 (2019) 1–11. doi:https://doi.org/10.1016/j.rcim.2019.05.009.
URL https://www.sciencedirect.com/science/article/pii/S0736584518303831

[21] M. Borish, A. Roschli, Ornl slicer 2.0: Towards a new slicing paradigm, in: 2021 International Solid Freeform Fabrication Symposium, University of Texas at Austin, 2021.

[22] Y. Jin, H. A. Pierson, H. Liao, Toolpath allocation and scheduling for concurrent fused filament fabrication with multiple extruders, IISE Transactions 51 (2) (2019) 192–208.

[23] J. Etienne, N. Ray, D. Panozzo, S. Hornus, C. C. L. Wang, J. Martínez, S. McMains, M. Alexa, B. Wyvill, S. Lefebvre, Curvislicer: Slightly curved slicing for 3-axis printers, ACM Trans. Graph. 38 (4) (jul 2019). doi:10.1145/3306346.3323022.
URL https://doi.org/10.1145/3306346.3323022

[24] S. Liu, T. Liu, Q. Zou, W. Wang, E. L. Doubrovski, C. C. L. Wang, Memory-Efficient Modeling and Slicing of Large-Scale Adaptive Lattice

Structures, Journal of Computing and Information Science in Engineering 21 (6), 061003 (05 2021). arXiv:https://asmedigitalcollection.asme.org/computingengineering/article-pdf/21/6/061003/6696935/jcise_21_6_061003.pdf, doi:10.1115/1.4050290. URL https://doi.org/10.1115/1.4050290

[25] P. Nayyeri, K. Zareinia, H. Bougherara, Planar and nonplanar slicing algorithms for fused deposition modeling technology: a critical review, The International Journal of Advanced Manufacturing Technology (2022) 1–26.

[26] L. J. Gibson, Cellular materials: Structure, properties, and applications, in: Frontiers of Engineering: Reports on Leading Edge Engineering from the 1996 NAE Symposium on Frontiers of Engineering, National Academies Press, 1997, p. 75.

[27] P. Dziewit, P. Platek, J. Janiszewski, M. Sarzynski, M. Grazka, R. Paszkowski, Mechanical response of additive manufactured regular cellular structures in quasi-static loading conditions—part i experimental investigations, in: Proceedings of the 7th International Conference on Mechanics and Materials in Design, 2017, pp. 1061–1074.

[28] M. Kucewicz, P. Baranowski, J. Małachowski, A. Popławski, P. Płatek, Modelling, and characterization of 3d printed cellular structures, Materials & Design 142 (2018) 177–189. doi:https://doi.org/10.1016/j.matdes.2018.01.028. URL https://www.sciencedirect.com/science/article/pii/S0264127518300364

[29] T. Liu, S. Guessasma, J. Zhu, W. Zhang, Designing cellular structures for additive manufacturing using voronoi–monte carlo approach, Polymers 11 (7) (2019) 1158.

[30] P. Song, Z. Fu, L. Liu, C.-W. Fu, Printing 3d objects with interlocking parts, Computer Aided Geometric Design 35 (2015) 137–148.

[31] M. Spoerk, F. Arbeiter, H. Cajner, J. Sapkota, C. Holzer, Parametric optimization of intra-and inter-layer strengths in parts produced by extrusion-based additive manufacturing of poly (lactic acid), Journal of applied polymer science 134 (41) (2017) 45401.

[32] M. Beecroft, 3d printing of weft knitted textile based structures by selective laser sintering of nylon powder, in: IOP Conference Series: Materials Science and Engineering, Vol. 137, IOP Publishing, 2016, p. 012017.

[33] I. A. Malik, M. Mirkhalaf, F. Barthelat, Bio-inspired "jigsaw"-like interlocking sutures: Modeling, optimization, 3d printing and testing, Journal of the Mechanics and Physics of Solids 102 (2017) 224–238.

[34] M. Ribeiro, O. S. Carneiro, A. F. da Silva, Interface geometries in 3d multi-material prints by fused filament fabrication, Rapid Prototyping Journal (2018).

[35] A. Dijkshoorn, V. Ravi, P. Neuvel, S. Stramigioli, G. Krijnen, Mechanical interlocking for connecting electrical wires to flexible, fdm, 3d-printed conductors, in: 2022 IEEE International Conference on Flexible and Printable Sensors and Systems (FLEPS), IEEE, 2022, pp. 1–4.

[36] I. Mustafa, T.-H. Kwok, Development of intertwined infills to improve multi-material interfacial bond strength, Journal of Manufacturing Science and Engineering 144 (3) (2022).

[37] A. Dyskin, Y. Estrin, A. Kanel-Belov, E. Pasternak, A new concept in design of materials and structures: Assemblies of interlocked tetrahedron-shaped elements, Scripta Materialia 44 (12) (2001) 2689–2694.

[38] A. V. Dyskin, Y. Estrin, A. J. Kanel-Belov, E. Pasternak, Toughening by fragmentation—how topology helps, Advanced Engineering Materials 3 (11) (2001) 885–888.

[39] A. V. Dyskin, Y. Estrin, E. Pasternak, H. C. Khor, A. J. Kanel-Belov, Fracture resistant structures based on topological interlocking with non-planar contacts, Advanced Engineering Materials 5 (3) (2003) 116–119.

[40] B. Zareiyan, B. Khoshnevis, Effects of interlocking on interlayer adhesion and strength of structures in 3d printing of concrete, Automation in Construction 83 (2017) 212–221.

[41] C. Duty, J. Failla, S. Kim, T. Smith, J. Lindahl, V. Kunc, Z-pinning approach for 3d printing mechanically isotropic materials, Additive Manufacturing 27 (2019) 175–184.

[42] T. Kuipers, R. Su, J. Wu, C. C. Wang, Itil: Interlaced topologically interlocking lattice for continuous dual-material extrusion, Additive Manufacturing 50 (2022) 102495.

[43] J.-C. Latombe, Robot motion planning, Vol. 124, Springer Science & Business Media, 2012.

[44] H. S. M. Coxeter, Twisted honeycombs, no. 4 in Conference Board on the Mathematical Sciences: Regional Conference Series in Mathematics, American Mathematical Society, 1970.

[45] H. S. M. Coxeter, Regular polytopes, Courier Corporation, 1973.

[46] B. Grünbaum, G. C. Shephard, Tilings with congruent tiles, Bulletin of the American Mathematical Society 3 (3) (1980) 951–973.

[47] L. Poudel, L. G. Marques, R. A. Williams, Z. Hyden, P. Guerra, O. L. Fowler, S. J. Moquin, Z. Sha, W. Zhou, Architecting the cooperative 3d printing system, in: International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, Vol. 83983, American Society of Mechanical Engineers, 2020, p. V009T09A029.

[48] H.-C. Wu, T.-C. T. Chen, Quality control issues in 3d-printing manufacturing: a review, Rapid Prototyping Journal 24 (3) (2018) 607–614.