Domain Decomposition and Space Filling Curves in Toolpath Planning and Generation

Bertoldi M., Yardimci M. A., Pistor C. M., Güçeri S. I.
University of Illinois at Chicago

ABSTRACT

Ever increasing computer performance, along with significant developments in CAD/ CAM technology and high precision digital motion controllers lead to rapid and significant developments in the field of rapid prototyping/fabrication. Together, these elements offer a wide range of possible approaches in the toolpath planning issue; two main sets have been analyzed and applied to Fused Deposition Modeling process.

Domain decomposition is a frequently used technique in computational methods. Within the context of present study, this approach is used to divide arbitrary layer geometries into smaller regions of simpler shape. A foreseeable advantage in such an approach is maximizing strength characteristics through thermal management. This is achieved by utilizing space filling curves which are mathematical entities that offer the possibility of building a wide range of structures, covering the surface of a single layer with one continuous curve. To evaluate the proposed concepts, ABS structures and ceramic green bodies have been successfully built.

OVERVIEW

The term toolpath planning is defined here as the collection of operations performed to provide the Rapid Prototyping hardware with the set of building commands.

Although computer-controlled machines are now commonly available, the algorithms for driving them automatically are still subject of research. In CNC machining for example, human intervention and ingenuity are still necessary to design and build the necessary jigs for clamping of the workpiece. The integration of RP equipment with PC-based controllers offers the possibility of using common software development tools and resources for toolpath planning.

Fused Deposition hardware is very similar to basic operation of pen-plotters and the machine level controllers are based on the same language. As a result, they can work more efficiently with patterns (i.e. porous structures) that have "polylines" as the basic geometric features, compared to the current mode of operation, which works on filled polygons (monolithic models).

The algorithms that have been developed and applied to the Fused Deposition can be considered in two main categories:

- Slice level manipulation
  - Domain decomposition is a common technique in computational methods, used here to divide arbitrary layer geometries into smaller regions of simpler shape.
- Machine level manipulation
  - Space filling curves are mathematical entities that offer the possibility of building a wide range of structures, covering the surface of a single layer with one continuous curve.

SLICE LEVEL MANIPULATION

In the current study domain decomposition was demonstrated using a Stratasys 1650 modeler. The standard Stratasys software (QuickSlice™) performs the translation from the 3D CAD model to the 2½D model through a slicing algorithm. At this point, the geometry of the part is described with a finite set of 2D layers, in order to match the SFF building paradigm. For certain applications, it can be more useful to directly generate or modify the geometry of the desired structure at slice level, i.e. in modeling of three dimensional objects from medical images where the original data is a set of 2D images. Furthermore, working on 2D contours allows the use of simpler and more common algorithms.

The slice information is contained in a SSL file, where it is described as a contour representing a closed planar surface. In the standard software processing, the slices are subsequently filled with a set of parallel lines, representing the roads. At the SSL level, the slice represents the building unit and it belongs to a different "group" to which the following properties can be assigned:

- Material
- Build style
- Contour (yes/no)
- Road width
- Air gap between roads
- Raster orientation, which is usually set to [+45 –45] for two consecutive levels.

The main constraint is the fixed nature of the raster, which is always made of straight parallel roads. This affects the physical structure of the part and all its characteristics, including mechanical properties, void contents, surface and bonding quality and accuracy. Furthermore, the topology of the parts that can be created is limited to standard solid structures.

A set of routines and programs has been developed in a combined FORTRAN/MATLAB environment, in order to explore the possibilities of the process.

DOMAIN DECOMPOSITION

Domain decomposition techniques are mostly used in computational analysis, when the mathematical model describing the physical phenomena cannot be solved in closed form. The idea is to decompose the 2 or 3-dimensional domain into regions of simpler geometry (usually polygons in 2D and polyhedra in 3D) and to discretize the equations on the obtained grid, in order to generate an approximate solution. This approach, which had its real boost with the advent of computers, has requested the development of more refined and accurate mesh generator algorithms, especially for unstructured grids.

In Rapid Prototyping, the physical domain can be assimilated to the part to be built in the 3D representation, where the computational domain is the solid model (STL file). However, most RP/SFF techniques are based on a layered manufacturing paradigm and the eventual domain decomposition of the STL model cannot be reproduced in the physical part. Instead of considering the entire solid model, the domains to be decomposed are the single layers and the final mesh is a new layer, with the same external contour but subdivided in a variable size and shape mesh.

With this approach, it is possible to increase the number and manipulate the characteristics of the domains, which at this level represent the building units. This results in a more flexible and customizable building process.

DELAUNAY TRIANGULATION

In two dimensions, a triangulation of a set V of vertices is a set T of triangles whose vertices collectively form V, whose interiors do not intersect each other, and whose union completely fills the convex hull of V.

One of the most common techniques for generating triangular grids is based on Delaunay triangulation. The Delaunay triangulation D of V, introduced by its author in 1934, is the graph defined as follows. Any circle in the plane is said to be empty if it contains no vertex of V in its interior (Vertices are permitted on the circle). Let u and v be any two vertices of V. The edge UV is in D if and only if there exists an empty circle that passes through U and V. An edge satisfying this property is said to be Delaunay.

Although initial Delaunay triangulation algorithms have been developed for simply connected convex polygons [1], [2], current algorithms [3], [4], are able to generate variable density Delaunay triangulations within multiply connected arbitrary geometry polygons; some of them find applications in numerical modeling problems [5]. Delaunay triangulation is perfectly suited for a domain described with a set of segments; in our case, the program *trian.exe* developed in 1996 by Shewchuk [6] is able to generate triangles at a rate of 10^5 points in 30 seconds. The structure of the algorithm is presented here in pseudo-code:

```
ssl_head()

for i=1:number_of_layers
      for j=1:number_of_slices
```

```
                read slice file
                translate current_slice -> input_domain
                call trian.exe
                read output file
                write new_slice
          end
    end
```

The function *ssl.m* has been implemented in MATLAB. The old set properties from the input SSL file generated with QuickSlice are read, and the file is scanned for number of layers and number of distinct domains within each of them. The structure of the layer is then read, translated and written into a file understandable by *trian.exe*. Once the domain decomposition has been performed, the generated triangular mesh is read, and each triangle is assigned to one of the previously set group (each of them can have different properties, as described above). Finally, the new data structure is written in SSL format and can be reentered in the standard software for toolpath generation.

In the tested cases, two different command-line option were used with *trian.exe:*

- *–p,* which causes the program to generate a Planar Straight Line Graph (PSLG), which is a collection of points and segments. Segments are edges whose endpoints are points in the PSLG, and whose presence in any mesh generated from the PSLG is enforced.

- *–a x,* which constrains the maximum area of a single triangle to *x*.

Each layer of the tensile test bars depicted in Figure 1 have been decomposed with a different maximum area constraint; the resulting triangles have been randomly assigned to a group with variable road orientation [0 ±30 ±45 ±60 90].

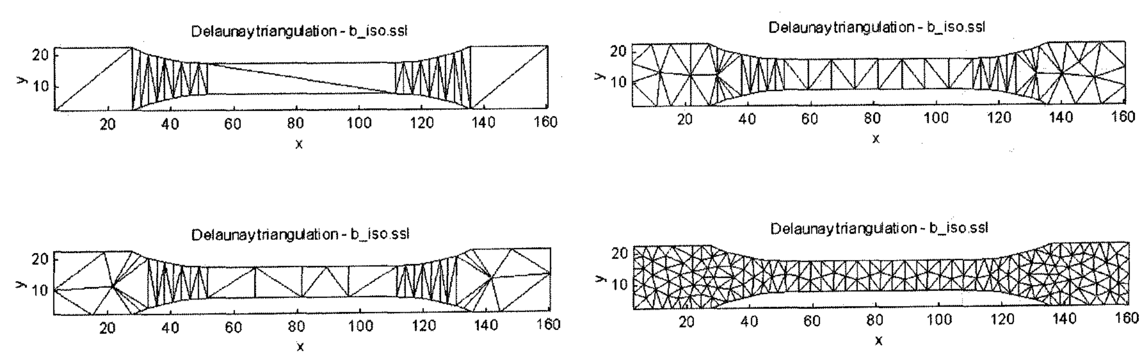

Figure 1. Domain decomposition of a tensile test bar

Another example is depicted in Figure 2. The interior of the octagon was tessellated with a constrained Delaunay triangulation, and constructed triangles were defined as individual contours within the SSL file. Road width was fixed to 0.5 mm; individual triangles were then randomly assigned to a set of group with variable raster angles and air-gaps (0.3 to 0.7 mm) end

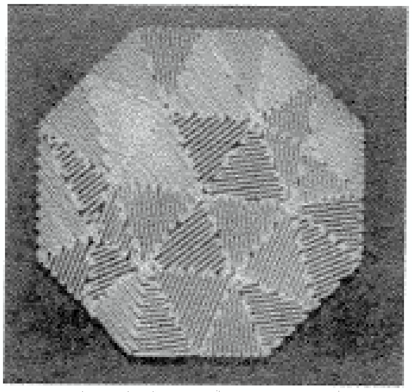

Figure 2. Domain decomposition structure

MACHINE FILE GENERATION

The only way to fully explore the possibilities of the FDM process is to directly generate the machine level language, in order to avoid all the constraints of the standard toolpath processing software. It is in fact possible to skip the solid model generation and slicing steps (STL – SSL files) and directly output the final machine file (SML).

The Stratasys FDM 1650 machine used for the experimental tests is driven by an Asymtek A-201 digital motion controller. The A-201 controls the *x-y* movement of the depositing head, the *z* movement of the stage, and the rotation of the two electrical servo-motors mounted on the head that feed the thermoplastic wire into the two liquefiers. The controller uses Automove Control Language (ACL) for programming [7]; Stratasys has implemented a slightly modified version of this language, called Stratasys Machine Language (SML). It is similar to Hewlett Packard's PCL used to control plotters and all commands are strings of ASCII text.

The structure of the SML file has been reverse-engineered in order to separate the common control sequences from the real flow and motion control commands. A set of routines has been developed in FORTRAN and then ported to MATLAB; the structure of the basic algorithm with a detailed description of each function is shown here in pseudo-code:

```
smlhead ()
purge ()

for i=1:number_of_layers_base
      base ()
      (curvebrk)
end

for i=1:number_of_layers
      level ()
      purge ()
      curve ()
      toolpath ()
      (curvebrk)
      endcurve ()
end
smltail ()
```

where:

`smlhead`	Establishes communication parameters, initializes the motion controllers and prepares the FD machine for the build.
`purge`	Performs on-the-fly cleaning operations (tip-wipe).
`level`	Switches materials if necessary and positions the z-stage for the current layer.
`curve`	Sets the flow-control parameters for the current road
`toolpath`	Generates motion control commands for the deposition of the road
`curvebrk`	Writes a pausing sequence. The motion controller has a very limited on-board memory that can store only 256 consecutive commands. When the buffer is full, the transmission is interrupted and the stored commands executed. Communication link is reactivated near the end of 256-command set and the next batch is loaded to the memory.
`smltail`	Performs final cleaning operations, clears the memory and puts the machine in stand-by mode for part removal.

It should be noticed that the 3D topology of the final structure can still be maintained; in fact, each layer can be generated with a different toolpath.

FRACTAL SPACE FILLING CURVES

A space-filling curve (SFC) is a continuous line that can cover a region of space without intersecting itself. Applications of space filling curves can be found in digital images processing, ray tracing optimization and fluid dynamics. The idea of using fractals in RP is due to the intrinsic nature of the curve: it can cover each layer without starting-stopping sequences. The length of the unit steps within SFC's can be set to a value equal to the width of the roads, in order to completely fill each layer.

The Hilbert space-filling curve (Figure 3) has been found particularly useful and consequently applied to the toolpath planning issue. The developed MATLAB program, shown in pseudo-code, generates the Hilbert curve, which is stored in a vector (toolpath), translated into SML language and written to file. The function `main()` contains the connectivity matrix (a [4x4x3] array) that defines the topology of the curve and recursively calls the `hilbc()` function:

```
main ()
      conn_matrix = [...]
      hilbc (order);
end (main)

function hilbc (order)

if (order>0)
      for n=1:order
            hilbc (order-1);
            generate_toolpath ();
      end
end
```

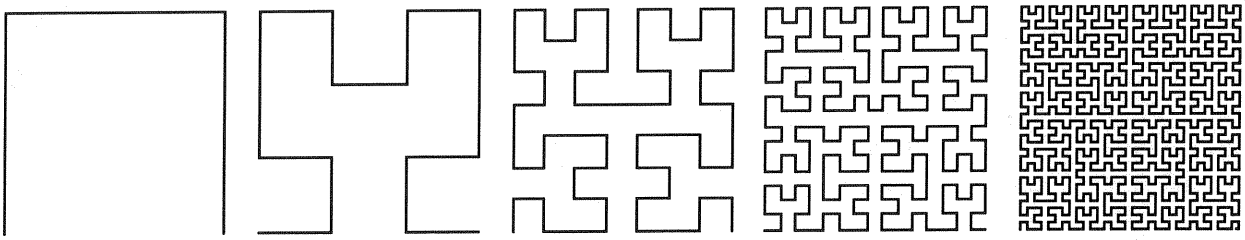

Figure 3. Hilbert curve generation

One of the main advantages of this approach is related to the thermal history of the part produced; from microscopy images, it has been observed that whenever the head deposits long straight lines, the bonding quality between two contiguous roads can be poor or non-existent. This is due to the thermal expansion and subsequent cooling and shrinking of the single road. With the fractal approach, it is possible to depose one layer reducing significantly the length of the movements of the head. In this way, the void contents can be lowered and the bonding quality can be improved. The main disadvantage is the build time, which is 2-4 times longer compared to the standard toolpath, due to the exponentially increased number of motion commands.

The microstructure generated by 0/-90° stacking of a 4th order Hilbert curve is shown in Figure 4-a; Figure 4-b shows the part produced using this approach (ABS material).

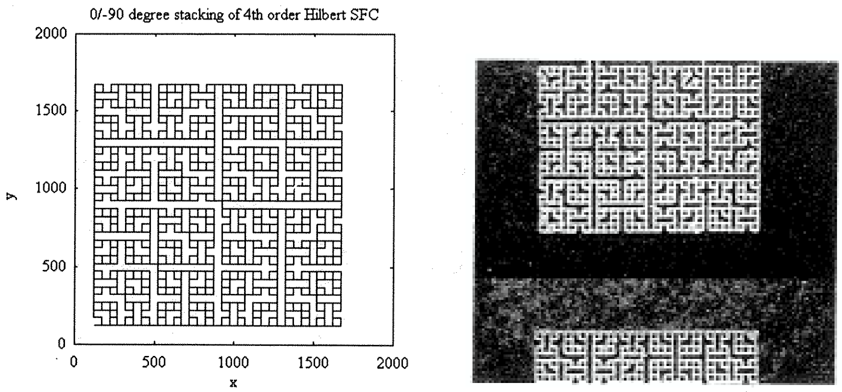

Figure 4. Microstructure toolpath (a) and real part (b)

APPLICATIONS

Possible applications of the proposed techniques cover all the Fused Deposition uses, where custom physical and mechanical properties of the parts must be achieved. The software routines developed can be adapted to different needs and are easily configurable and customizable.

In particular, one interesting application concerns the mechanical behavior of the FDM parts. The standard raster settings cover every slice with a set of parallel roads, causing a strongly orthotropic behavior of the obtained part. Decomposing the domain and assigning the properties to each new contour independently allows more control on the internal microstructure. Specimens built with fractal space filling curves were tested, showing better mechanical properties compared to the standard parts.

Another interesting application taking direct advantage of the proposed techniques is in porous structure generation algorithms; see [8] for more details.

CONCLUSIONS

With the new programming tools available, it is possible to generate different routines for developing new toolpath planning techniques. Some of them have been presented, discussed and deemed effective in the toolpath planning issue, either as a stand-alone product or interfaced with the standard processing software. Future work will include integration with other domain decomposition algorithms and STL three-dimensional domain decomposition.

REFERENCES

1. Bowyer A., "Computing Dirichlet tessellations," *The Computer Journal*, vol. 24 no. 2, pp. 162-166, 1981.
2. Watson D. F., "Computing the n-dimensional Delaunay tesselation with application to Voronoi polytopes," *The Computer Journal*, vol. 24, no. 2, pp. 167-171, 1981.
3. Guibas L. J., Knuth D. E., Sharir M., "Randomized incremental construction of Delaunay and Voronoï diagrams," *Algorithmica*, vol. 7, no. 4, pp. 381-413, 1992.
4. Shimada K., Physically-Based Mesh Generation: Automated Triangulation of Surfaces and Volumes via Bubble Packing, Ph.D. thesis, Massachusetts Institute of Technology, 1993
5. M. A. K. Posenau, D. M. Moun t., "Delaunay triangulation and computational fluid dynamics meshes", National Aeronautics and Space Administration, Langley Research Center Hampton, V 1992.
6. Jonathan Richard Shewcuk , "Delaunay Refinement Mesh Generator" Computer Science Department, PhD Thesis, Carnegie Mellon University 1997.
7. Automotive Control Language (ACL) Programming Reference Manual - Version 3.61 Asymtek 1996.
8. M. Bertoldi, M. A. Yardimci, C. Pistor, S. I. Güçeri, S. C. Danforth, "Porous Structure Generation with Fused Deposition", Proceedings of the 1998 Solid Freeform Fabrication Symposium, The University of Texas at Austin, Austin, Texas 1998.