Multiprocessing and Correction Algorithm of 3D-models for Additive Manufacturing

R R Anamova, S V Zelenov, M U Kuprikov, A V Ripetskiy
Moscow Aviation Institute (National Research University),
Moscow, Russia

E-mail: anamova.rushana@yandex.ru

Abstract. This article addresses matters related to additive manufacturing preparation. A layer-by-layer model presentation was developed on the basis of a routing method. Methods for correction of errors in the layer-by-layer model presentation were developed. A multiprocessing algorithm for forming an additive manufacturing batch file was realized.

1. Introduction

The emergence of new 3D-printing technologies has resulted in a dramatic increase in hardware and software development [1-3]. Printing capabilities, material specifications and accuracy of modern industrial additive manufacturing equipment, all point at the feasibility of using such technologies for production of different engineering items. At the same time, additive manufacturing preparation software remains an open issue, due to the fact that modern additive manufacturing preparation software is not multipurpose [4].
Software systems for additive manufacture preparation have the following weaknesses:

- costs of material and computing resources;
- impossibility to select algorithms for error correction;
- impossibility of editing, control of sintering layers and checking of item topology after slicing;
- high probability of errors during files transfer from working chamber arrangement software to the machine control program generating software;
- impossibility of working with operating systems other than Windows and other hardware platforms.

In this connection, the development of new additive manufacturing preparation software remains a pressing problem.
Preconditions for the development of additive manufacturing preparation software include:

- formulation of requirements and identification of constraints for automation tools of layer-by-layer synthesis method for manufacture preparation;
- definition of mathematical and algorithmic support;
- verification of the developed mathematical support and software.

2. Development of layer-by-layer model presentation

During manufacture preparation process, discrete presentation of the model with 3D-printer precision is generated. We can describe the model surface as a set of triangles which are identified by three ordered vertices.
Empirically, the following model validity criteria were formulated:

- the surface should be continuous;

- the surface should confine the space;
- volumes confined with various surfaces should not overlap;
- different surfaces should not contain the same points.

Validity criteria may be checked using the routing method.

2.1. Routing method

Geometrical model is presented as a grid or a number of connected bounds (Figure 1). We can choose any point of space $\tilde{o}_0$ and vector $\overline{p}$ and consider intersection points of the resulting line and model bounds.

Let us consider intersection points for line $L(\overline{p}, x_0)$ and model bounds. Let us assume that $\overline{v_i}$, $i = [1...n]$ are bounds, which are intersected by line $L(\overline{p}, x_0)$ in points $\tilde{o}_i$. Since bound $\overline{v_i}$ is an ordered set of points, we can calculate a normal to the bound as a cross product:

$\overline{v_i} = [x_1, ..., x_3]$, $\overline{p_i} = [\overline{p_{i1}}, \overline{p_{i2}}] = [x_2 - x_1, x_3 - x_1]$.

All bounds intersected with line $L(\overline{p}, x_0)$ can be divided into exterior and interior ones. Bound $\overline{v_i}$ will be exterior for vector $\overline{p}$, if the angle between $\overline{p_i}$ and $\overline{p}$ is less than 90º. We can define it as a cross product:

$$v_i = \begin{cases} exterior, if \left(\overline{p_i}, \overline{p}\right) = \left|\overline{p_i}\right|\left|\overline{p}\right|\cos(\phi) > 0, \\ interior, if \left(\overline{p_i}, \overline{p}\right) = \left|\overline{p_i}\right|\left|\overline{p}\right|\cos(\phi) < 0 \end{cases}$$

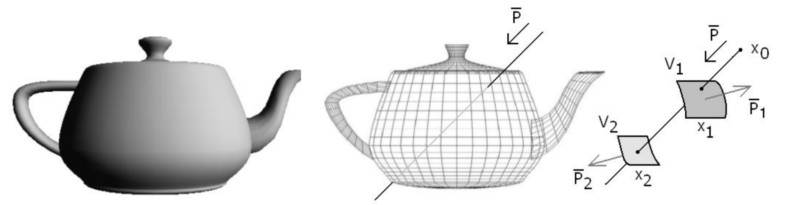


Figure 1. Model (left), grid presentation of the model (center) and routing (right)

Let us compose set Ì from pairs $(x_i, \overline{p_i})$, $\tilde{o}_i$ is intersection point of bound $\overline{v_i}$ with line L, $\overline{p_i}$ is normal vector of bound $\overline{v_i}$. Points $\tilde{o}_i$ are ordered on line L. Model is closed relatively to line L, if the first bound is exterior and orientation of each following bound is different from that of the previous bound. Using scalar product: a set of Ì $= \left(x_i, \overline{p_i}\right)$ pairs, $i = 1, ..., n$ is valid, if $(\overline{p_0}, \overline{p}) > 0$, $(x_0, x_i) < (x_0, x_{i+1})$, $(\overline{p_i}, \overline{p}) = -(\overline{p_{i+1}}, \overline{p})$.

Validity of the model as a whole may be formulated as follows: model V is valid, if no $\overline{p}$ vector and $\tilde{o}_0$ point exist, for which the set of Ì $= \left(x_i, \overline{p_i}\right)$ pairs, $i = 1, ..., n$ is not valid. We should check validity of the model only for the direction, from which its layer-by-layer presentation was calculated, and with resolution, which is necessary for manufacturing using particular equipment.

2.2. Generation of layer-by-layer model presentation

Layer-by-layer presentation of model V_i can be obtained using its subset in discrete space Z^3. Discrete model presentation (Figure 2) is based on function R: $R(V_i) \subset Z^3$. Using set of $M\left(x_{ij}, \overline{p}\right)$ pairs, which

IOP Conf. Series: Materials Science and Engineering **140** (2016) 012003 doi:10.1088/1757-899X/140/1/012003

result from intersection of the model with $\overline{p}$ line and contain point x_{ij}, we can form function R, which maps a set of M pairs in discrete space.

Let us define D_i as model V_i presentation in discrete space of whole numbers: $D_i = (d_1, ..., d_n), d_i \in Z^3, D_i \subset Z^3$.

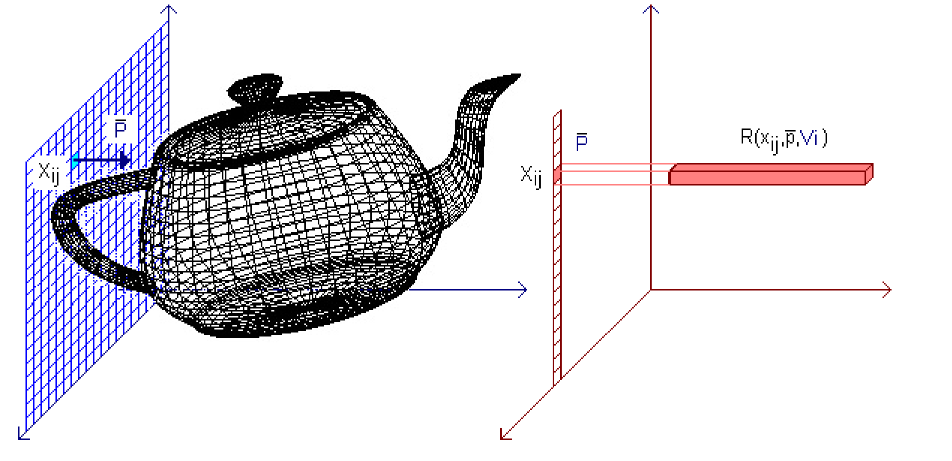


Figure 2. Parametrical and discrete model presentation

Discrete model presentation is used as input information for generation of numerical control commands; it can be obtained using the routing method for the set of x_{ij} pairs.

$$D(V_i) = \sum_{i,j} R\left(x_{i,j}, \overline{p}, V_i\right), D(V_i) \subset Z^3.$$

3. Development of methods for correction of errors in layer-by-layer model presentation

All additive manufacturing technological processes use layer-by-layer model presentation, which is realized through routing methods. When we use routing method algorithms, the input information includes discrete model presentation (triangles); the output information is layers. 'Layer' means vector or raster presentation of a model section in selected scanning plane.

Vector presentation can be obtained analytically by searching model sections in selected scanning planes. Raster presentation is a result of repetitive application of model triangles and scanning vector intersection points searching operation. Searching operation is repeated until the result with necessary precision is found.

The result of model routing is an ordered set of coordinates of the intersection point and normals in each point:

$$P(V_i) = [x_i, N_i], i = 1...n, x_i > x_{i-1}.$$

Using scanning vector direction and face normal, we can define the face type (outer or back) by scalar product sign. If we know outer and back faces of the model, we can identify errors in the geometry of a 3D-model.

In routing method, geometrical model contains no errors, if the ordered set of intersection points and appropriate normals contains first outer face, last back face, and faces interchange.

For example, the following set of points and normals does not contain routing errors (Figure 3) ($x_i < x_{i+1}$):

$$\{x_1, outer\}, \{x_2, back\}, \{x_3, outer\}, \{x_4, back\}.$$

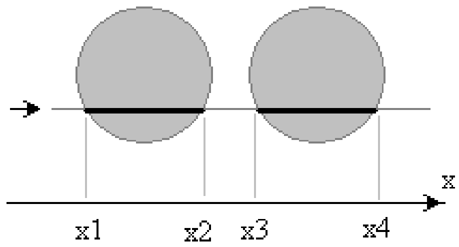


Figure 3. **S**et of points and normals

When analyzing interchange order for outer and back faces, we can reveal embedded surfaces or intersected surfaces of the geometrical model. Based on these results, we can formulate a condition: geometrical model contains no gross routing errors, if the ordered set of intersection points and appropriate normals contains first outer face, last back face, and the quantities of outer and back faces match.

For example, for embedded or intersected objects (Figure 4), order of faces will be as follows: $\{x_1, outer\}, \{x_2, outer\}, \{x_3, back\}, \{x_4, back\}$.

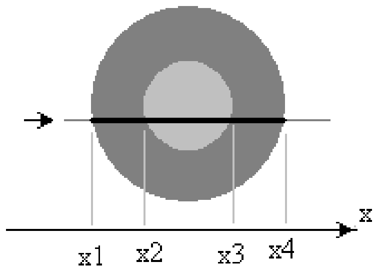


Figure 4. Embedded objects.

If the numbers of outer and back faces do not coincide, it is due to incomplete surfaces. For example, the following set (Figure 5) shows us errors in model geometry, which result in cavities in the end product.

$\{x_1, outer\}$.

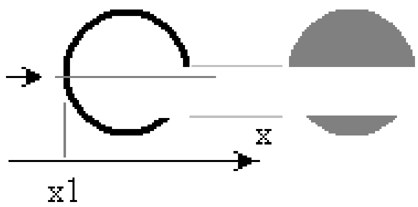


Figure 5. Model with geometry and layer-by-layer errors

Such errors can be corrected using a complicated routing method algorithm. If we save the history of previous routing results and restore the data, we will avoid the emptiness in presentation of surfaces and improve layer-by-layer model presentation. For example, if we save previous scanning data when the scanning vector goes bottom-up, we will be able to complete the surface with the emptiness error (Figure 6).

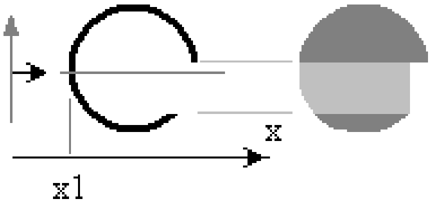
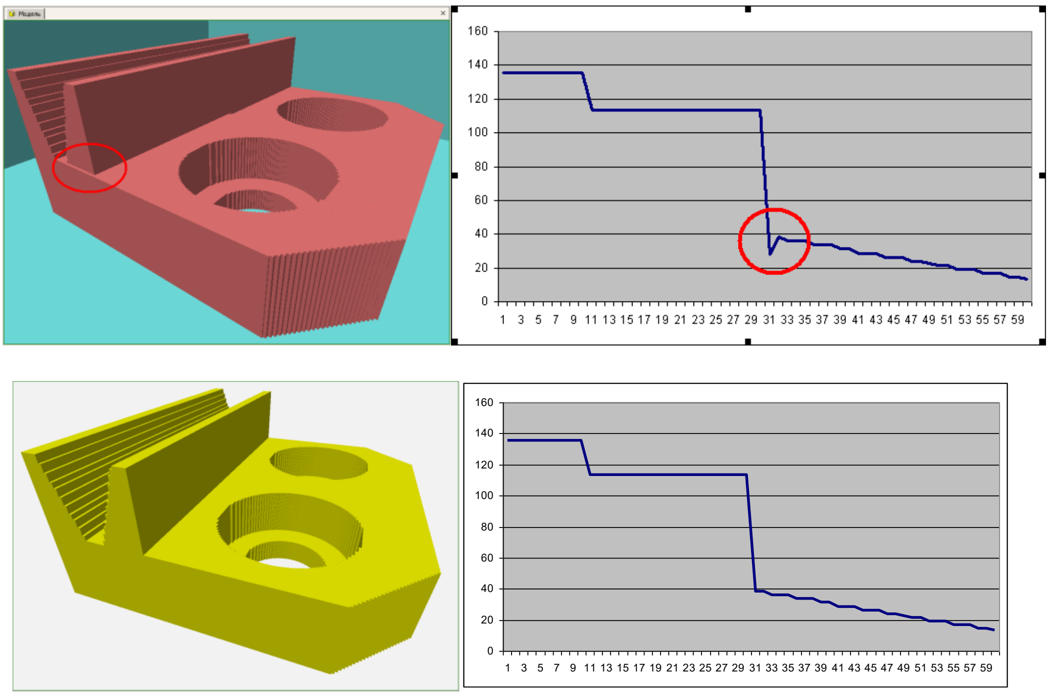


Figure 6. Restoration of model with emptiness error

The developed method is realized in the Program for Additive Manufacture Preparation (PAMP) [4]. Based on discrete model presentation in STL format, layer-by-layer model presentation is realized with necessary precision. Generated layers can be exported in raster format (BMP files) or can be used for G-code generation.

PAMP analyzes routing results for layer-by-layer presentation and checks for model geometry errors. Then, found geometry errors can be corrected automatically using the developed algorithm (Figures 7a and 7b). In Figure 7a, the geometry error and its layer position in the graph is marked red.



a)

b)

Figure 7. Correction of 3D-model geometry error: a) layer-by-layer representation of a 3D-model with geometry error (on the right is the dependence of layer volume on height); b) corrected geometry of the model (on the right is a new dependence of layer volume on height)

4. Development of the multiprocessing algorithm for batch file forming

Every additive manufacturing process is based on batch file, which is to be executed by 3D-printing equipment. The batch file contains commands for the movement of a printing jet or laser beam (depends on 3D-printing technology).

The role of the batch file in additive manufacturing is shown in Figure 8.

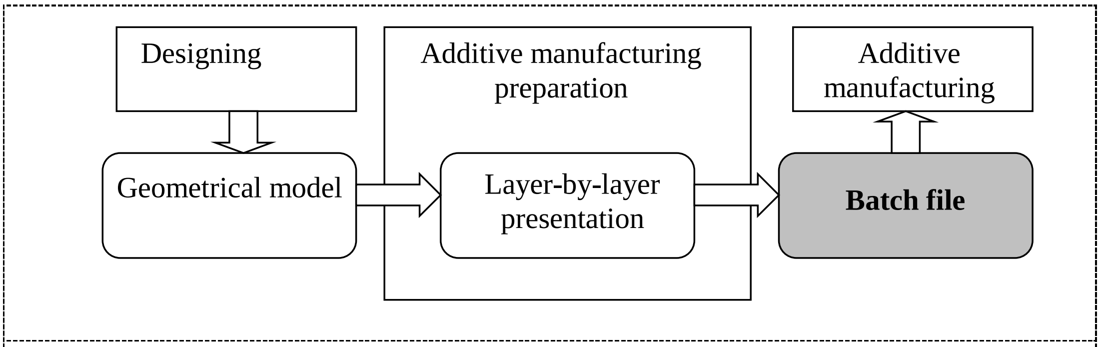
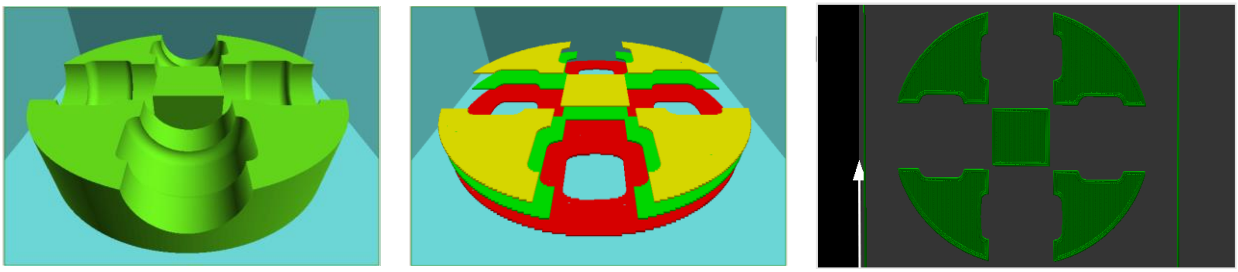


Figure 8. Batch files in additive manufacturing

Most 3D-printers use standard G-Code format as the batch file format. The batch file in G-code format is a text file, which contains setting commands and ordered commands of blank cycle (G0) and working stroke (G1). Example: G1 X48.700 Y1.049 E2.31026 F1350.000.
Geometrical model (a), layer-by-layer presentation (b) and visual presentation of the batch file (c) are shown in Figure 9.

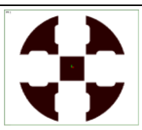
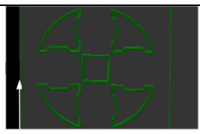
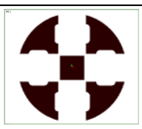


(a) (b) (c)

Figure 9. Model presentation stages: (a) geometrical model; (b) layer-by-layer presentation; (c) visual presentation of the batch file

Input data for batch file generation contain layer-by-layer model presentation. Layer can be presented as a set of parametric contours or as a raster BMP file.
Command presentation has two sections: perimeter and filling. It is necessary for edge smoothness and for interior content.
Perimeter section requires continuous closed contour bypass. For filling section, it is necessary to use closed contour filling. The presentation algorithms are shown in Table 1.

Table 1. Presentation algorithms

Input data	Result	Algorithm type
 Plane layer	 Command presentation of perimeter	Continuous closed contour bypass
 Plane layer	 Command presentation of filling	Continuous closed contour filling

The authors developed algorithms for batch files in G-Code format (PAMP). Developed software components were adapted for selective laser sintering.

The scheme for paralleling batch files is presented in Figure 10.

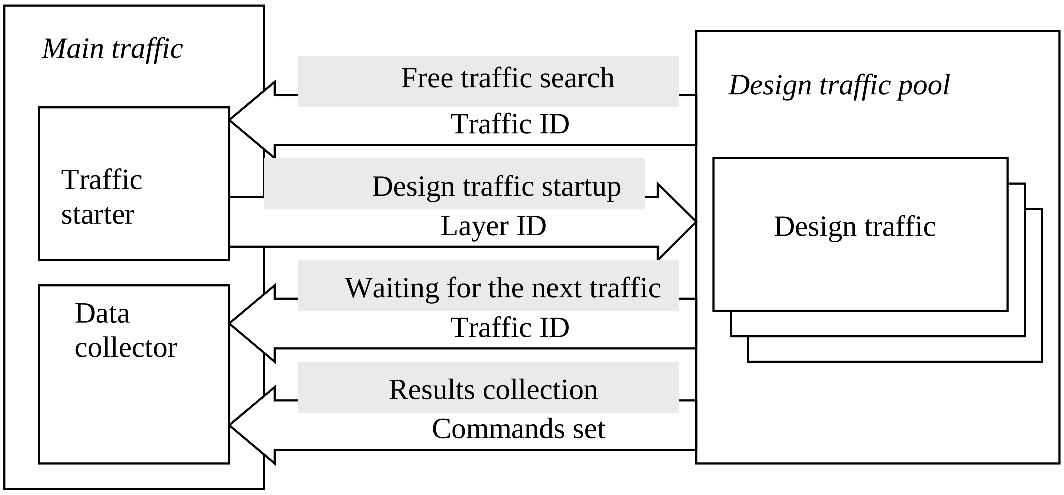


Figure 10. Batch files paralleling scheme

5. Conclusion

In Moscow Aviation Institute, new software for additive manufacturing preparation was developed—PAMP—with funding from the Russian Federal Targeted Program "Research and Development in Priority Fields of the Scientific and Technological Complex of Russia for 2014-2020", within the framework of arrangement no. 2.2, agreement no. 14.586.21.0019. PAMP is designed for G-code generation for additive manufacturing equipment and can be used for Selective Laser Sintering (SLS) and Fused Deposition Modelling (FDM). The main benefits of the developed software are:

1) PAMP allows user to check 3D-model geometry for errors and correct them before additive manufacturing preparation;
2) it is based on multiprocessing algorithms, which allows to speed up the additive manufacturing preparation process.

PAMP was tested during realization of real additive manufacturing tasks using FDM-technology and it have demonstrated good results: preparation of tested models for 3D-printing was successful, errors in 3D-models geometry were found and corrected, quality of produced models using PAMP-generated G-code was acceptable. Generation of G-code using developed multiprocessing algorithm was faster than that using single-processor algorithm according to linear relation.

References

[1] Dongdong Gu 2015 *Laser additive manufacturing of high-performance materials* (Berlin: Springer)
[2] Gibson, Rosen, Stucker 2010 *Additive manufacturing technologies*. (Berlin: Springer)
.
[3] Patri K. Vevuvinod, Weiyin Ma 2004 *Rapid prototyping. Laser-based and other technologies* (Berlin: Springer)
[4] Anamova R, Avtushenko A, Ivanov A, Ripetskiy A, Osipov A 2015 *J.Vestnik BGTU*. **2** (46) 8