Journal of Manufacturing Systems 30 (2011) 154–164

Contents lists available at ScienceDirect

Journal of Manufacturing Systems

journal homepage: www.elsevier.com/locate/jmansys

Technical paper

Adaptive tool-path generation of rapid prototyping for complex product models

G.Q. Jin [a], W.D. Li [a,*], C.F. Tsai [b], L. Wang [c]

[a] Faculty of Engineering and Computing, Coventry University, UK
[b] Department of Industrial Management & Enterprise Information, Aletheia University, Taiwan
[c] Virtual Systems Research Centre, University of Skövde, Sweden

ARTICLE INFO

Article history:
Received 15 January 2011
Received in revised form 12 May 2011
Accepted 16 May 2011
Available online 9 August 2011

Keywords:
Rapid prototyping
Tool-path generation
Adaptive algorithm

ABSTRACT

Rapid prototyping (RP) provides an effective method for model verification and product development collaboration. A challenging research issue in RP is how to shorten the build time and improve the surface accuracy especially for complex product models. In this paper, systematic adaptive algorithms and strategies have been developed to address the challenge. A slicing algorithm has been first developed for directly slicing a Computer-Aided Design (CAD) model as a number of RP layers. Closed Non-Uniform Rational B-Spline (NURBS) curves have been introduced to represent the contours of the layers to maintain the surface accuracy of the CAD model. Based on it, a mixed and adaptive tool-path generation algorithm, which is aimed to optimize both the surface quality and fabrication efficiency in RP, has been then developed. The algorithm can generate contour tool-paths for the boundary of each RP sliced layer to reduce the surface errors of the model, and zigzag tool-paths for the internal area of the layer to speed up fabrication. In addition, based on developed build time analysis mathematical models, adaptive strategies have been devised to generate variable speeds for contour tool-paths to address the geometric characteristics in each layer to reduce build time, and to identify the best slope degree of zigzag tool-paths to further minimize the build time. In the end, case studies of complex product models have been used to validate and showcase the performance of the developed algorithms in terms of processing effectiveness and surface accuracy.

1. Introduction

Rapid prototyping (RP) has been identified as an innovative manufacturing technology in recent years [1]. It provides a useful means to support quick and cost-effective model verification so as to facilitate product development collaboration. Different from conventional forming and machining processes, the RP technology is an additive build process by accumulating materials, layer-by-layer, as specified by a Computer-Aided Design (CAD) system. Based on this working principle, various RP machines have been developed, including Stereolithography Apparatus (SLA), 3D Printing (3-DP), Fused Deposition Modelling (FDM), Selective Laser Sintering or Melting (SLS/SLM), Laminated Object Manufacturing (LOM), etc. The distinguished advantage of RP is that it is a mould-less process, suitable for free-form and complex geometry product realization such as biomedical product models [2]. However, some challenging problems existed in RP include poor surface accuracy and extensive fabrication process especially for complex product models. There is a strong desire to reduce build time and enhance surface accuracy significantly through improved software, hardware and material utilization in RP to better support design verification and complex product models fabrication [3].

Process planning is a critical stage in RP, and a good process planning can enable the efficient and accurate manufacturing of complex product models [4]. Orientation determination, support structure determination, slicing and tool-path generation are four essential steps in process planning.

(1) *Orientation determination*. Different build orientation for a CAD model will affect the surface quality, the build time, the complexity of support structure and the total number of sliced layers. The build orientation is usually determined based on the height, surface quality requirement, mechanical properties and external support structure requirement of a CAD model [5,6];
(2) *Support structure determination*. It is used to support the build materials when CAD models have hollow or overhanging structures, and different designs of the structure affect the build time, surface quality and mechanical properties of the models. Some software packages have been developed in this research field to automatically generate the support of RP fabrication, such as Materials' Magics RP and Marcam Engineering's VisCAM RP, etc. [7,8];

* Corresponding author. Tel.: +44 2476 88 8940.
E-mail address: weidong.li@coventry.ac.uk (W.D. Li).

0278-6125/$ – see front matter.
doi:10.1016/j.jmsy.2011.05.007

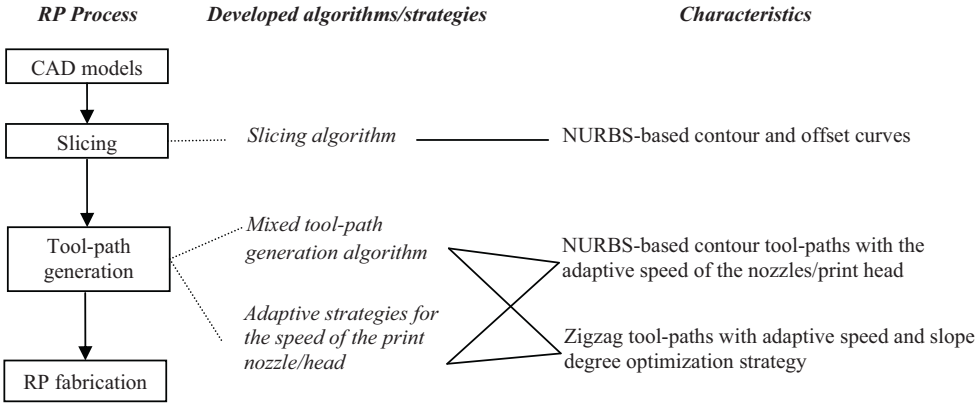


Fig. 1. Developed adaptive algorithms for RP tool-path generation.

(3) *Slicing*. It is to transfer a 3D CAD model to a series of layers. The developed methods can be divided into uniform slicing and adaptive slicing. In uniform slicing, the distance between two consecutive layers is the same, while in adaptive slicing, the distance between two layers varies depending on the surface curvatures of a CAD model [9]. Most of the RP systems use uniform slicing while research is being conducted to explore adaptive slicing;
(4) *Tool-path*. It is the trajectory of the nozzle/print head during a RP process to fill the interior of each layer. A tool-path strategy includes the determination of the topology, geometry and process parameters [9]. Several types of tool-path strategies and algorithms such as zigzag, contour, spiral and partition patterns have been developed with different considerations of the build time, cost, surface quality, warpage, strength and stiffness of RP models. A well-developed tool-path strategy can significantly optimize the time of fabrication, surface quality and mechanical properties of RP models.

In process planning, there are a number of elements affecting the surface quality and build time. This paper focuses on the improvement of the slicing approach and tool-path generation, which are the major technical challenge in RP process planning.

Stereolithography (STL) is a de facto data representation standard to support RP. However, it is inherently inaccurate in geometrical representation and needs much more storage spaces to represent a highly complex model. In addition, there are some problems in STL files such as gaps, holes, missing/degenerated/overlapping facets, etc. Recently, research has been carried out to develop directly slicing algorithms on a CAD model instead through a STL file conversion for tool-path generation to better support RP [10–14]. The advantage of directly slicing method is that it is able to maintain the accuracy of RP by using original CAD models and information. A tool-path is the trajectory of the nozzles/print heads in a RP process to fill the boundary and interior of each sliced layer. It is an important factor to determine the surface quality, strength and stiffness of a building model, and efficiency of the RP process. Various types of tool-path patterns have been developed for RP, such as zigzag [15,16], contour/spiral [17–19] and some space filling curves [20]. Based on the potentials in different conditions, research is expected to develop an adaptive and hybrid strategy to leverage the various tool-path generation strategies for complex product models. Build time is an important criterion of the RP technology. Some build time prediction models and optimization strategies have been developed [21–24]. Innovative and adaptive strategies are imperative to develop a more suitable and sensible build time mathematical model to calculate and further reduce the totally build time during the RP processes of complex product models.

In this paper, systematic adaptive algorithms/strategies to optimize the surface accuracy and build time of complex product models have been developed. The developed algorithms/strategies include:

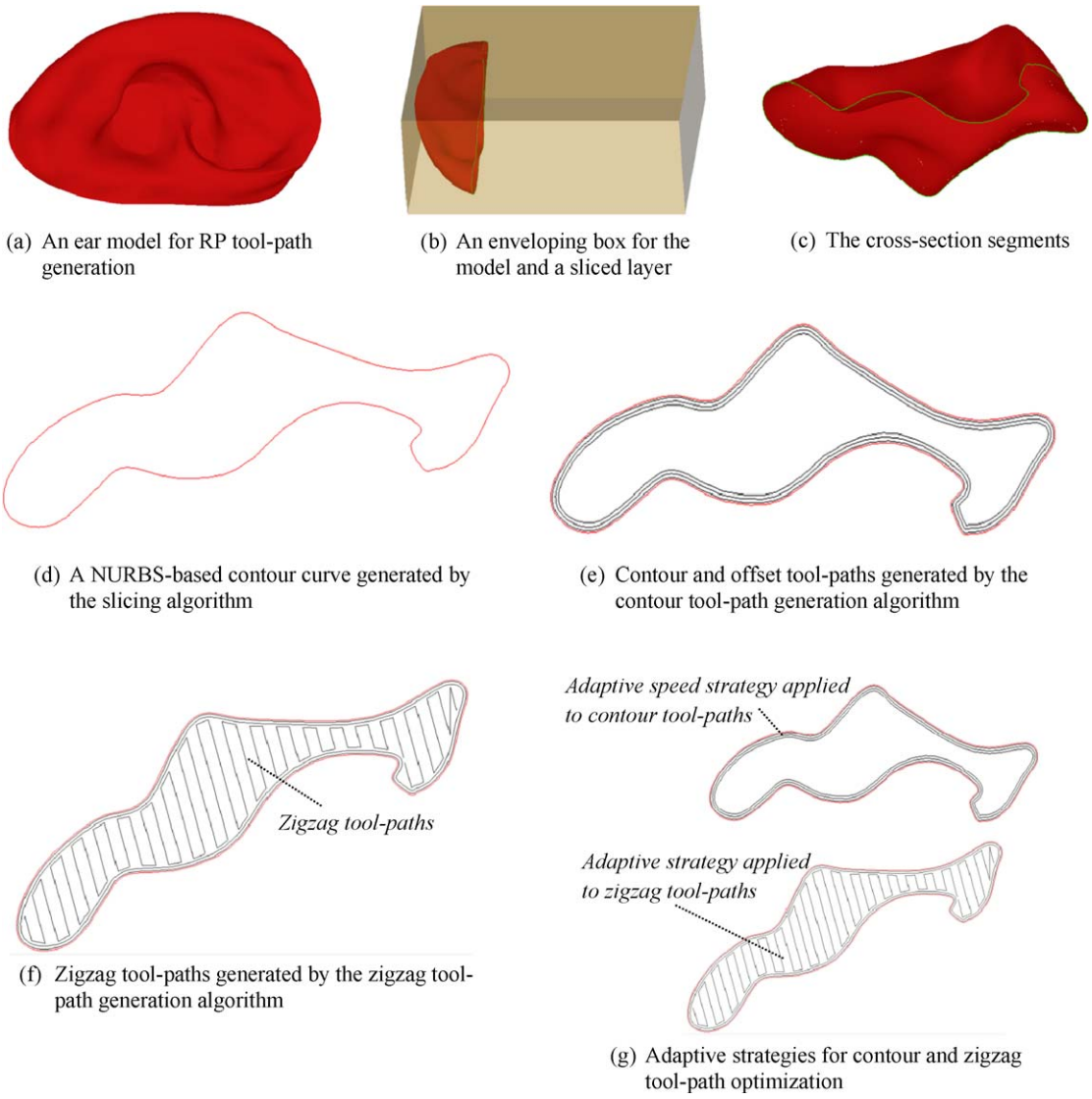
(1) Based on the Non-Uniform Rational B-Spline (NURBS) representation, contour and offset curves along the boundary of a RP model have been modelled for tool-path generation to improve the surface quality of RP. Meanwhile, based on a developed build time analysis model of contour tool-paths, a strategy has been designed to control the speed of RP nozzles/print heads adaptively for the various geometries of contours so as to optimize build time;
(2) Meanwhile, zigzag tool-paths for the internal area of the models are generated to simplify computing and fabrication processes. During the process, based on a developed build time analysis mathematical model of zigzag tool-paths, another adaptive strategy to determine the slope degree of zigzag tool-paths leading to the least build time has been devised.

In the end, two case studies for complex RP models have been used to demonstrate the performance of this research in terms of processing efficiency and surface accuracy.

2. Methodology and characteristics

The overview of the developed methodology is presented as follows.

(1) A slicing algorithm has been developed to slice a CAD model instead of a STL model for RP tool-path generation. After slicing, a closed NURBS curve is generated to represent the boundary contour of each sliced layer to maintain the accuracy of the original CAD model;
(2) A mixed tool-path algorithm has been developed to generate contour and zigzag tool-paths to meet both the accuracy and efficiency requirements. Contour tool-paths are used to fabricate the boundary of each sliced layer to improve the surface quality of CAD models; zigzag tool-paths are to fabricate the interior area of the models to improve efficiency;
(3) Build time analysis mathematical models for contour and zigzag tool-paths have been formulated. Based on the models, two adaptive strategies have been designed to address different geometrical characteristics of CAD models and to achieve minimum build time, including a strategy to optimize the speed of



(a) An ear model for RP tool-path generation
(b) An enveloping box for the model and a sliced layer
(c) The cross-section segments
(d) A NURBS-based contour curve generated by the slicing algorithm
(e) Contour and offset tool-paths generated by the contour tool-path generation algorithm
(f) Zigzag tool-paths generated by the zigzag tool-path generation algorithm
(g) Adaptive strategies for contour and zigzag tool-path optimization

Fig. 2. An example to illustrate the developed methodology. (a) An ear model for RP tool-path generation; (b) an enveloping box for the model and a sliced layer; (c) the cross-section segments; (d) a NURBS-based contour curve generated by the slicing algorithm; (e) contour and offset tool-paths generated by the contour tool-path generation algorithm; (f) zigzag tool-paths generated by the zigzag tool-path generation algorithm; (g) adaptive strategies for contour and zigzag tool-path optimization.

the RP nozzle/print head along the contour tool-paths, and a strategy to obtain the optimized slope degree and the speed of the RP nozzle/print head along the zigzag tool-paths.

The developed algorithms and strategies in a RP process and the related characteristics are shown in Fig. 1. In Fig. 2, the methodology is illustrated for an ear model for RP tool-path generation. In the following sections, the details of the methodology are disclosed as two parts – in Section 4, the contour related algorithms and strategy, including the slicing algorithm for contour curve generation, the contour tool-path generation algorithm and the adaptive speed strategy of the RP nozzle/print head along contour tool-paths, are elaborated. In Section 5, the zigzag related algorithm and strategy, including zigzag tool-path generation algorithm and adaptive strategy of the RP nozzles/print heads to obtain the best build slope degree and the speed along the zigzag tool-paths, are explained in details.

3. Generation of contour-based tool-paths

3.1. NURBS-based contour curve and tool-path representation

The longest of the enveloping box for a CAD model can be determined as the direction of slicing. A uniform slicing strategy is applied to the model, and the set of intersected points on the boundary between the sliced layer and the model are used to build up a series of NURBS-based contour curve. A general form of a NURBS-based contour curve is represented below.

$$C^{i,j}(u) = \sum_{i=0}^{n} w_i N_{i,p}(u) C_P^{i,j} \tag{1}$$

where $C^{i,j}$ represents the jth contour curve in the ith RP layer; u is the parametric variable ($u = [0, 1]$); w_i is the weight associated with control points; $C_P^{i,j}$ is the control point; p is degree; and

$$N_{i,0}(u) = \begin{cases} 1 & u_i \le u \le u_{i+1} \\ 0 & \text{otherwise} \end{cases}, \quad \text{and} \quad N_{i,p}(u) = \frac{u - u_i}{u_{i+p} - u_i} N_{i,p-1}(u) + \frac{u_{i+p+1} - u}{u_{i+p+1} - u_{i+1}} N_{i+1,p-1}(u) \tag{2}$$

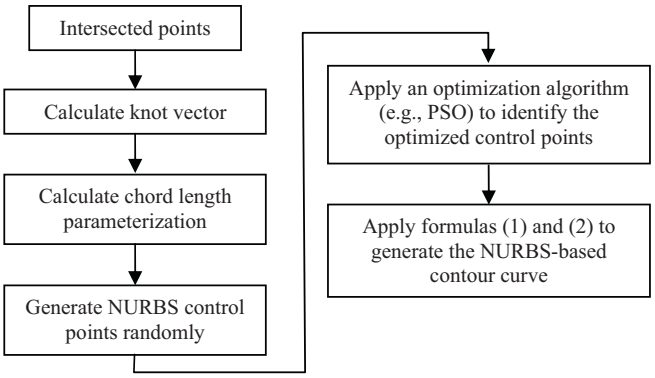


Fig. 3. The flow of generating the NURBS-based contour curve.

In order to establish a NURBS-based contour curve on the boundary between the sliced layer and the model ($C^{i,1}$), a fitting algorithm based on the intersected points has been used here [25]. A flow of the algorithm is shown in Fig. 3.

The generated contour curves can be further classified into two types defined as follows.

Definition 1. A control point $C_P^{k+1,1}$ and its two neighbour control points $C_P^{k,1}$ and $C_P^{k+2,1}$ form a triangle. Choose a random point P on the boundary or inside of the triangle, and draw a line to pass it. If the line only has two intersected points with the box formed by all the control points, and P is between the two intersected points, the control point $C_P^{k+1,1}$ is defined as convex; otherwise, it is concave.

Definition 2. If all the control points of a NURBS-based contour curve are convex, the curve is defined as a *Type I* contour curve. Otherwise, the curve is a *Type II* contour curve.

Two examples for a *Type I* contour curve and a *Type II* contour curve respectively are illustrated in Fig. 4.

3.1.1. Offset curve generation for a Type I contour curve

Based on the central point of the control points of the above generated contour curve, each control point for a new NURBS-based curve that offsets the contour curve (i.e., offset contour curve) is computed in the following.

$$C_P^{k,j} = D + \alpha(C_P^{k,1} - D) \tag{3}$$

where $C_P^{i,j}$ represents an offset contour curve in the ith layer; D represents the geometric central point of the control points of the contour curve; α is the ratio to generate the new control points of the offset contour curve; α is determined by the diameter of the RP nozzles/print heads and the overlapping rate between two neighbour tool-paths.

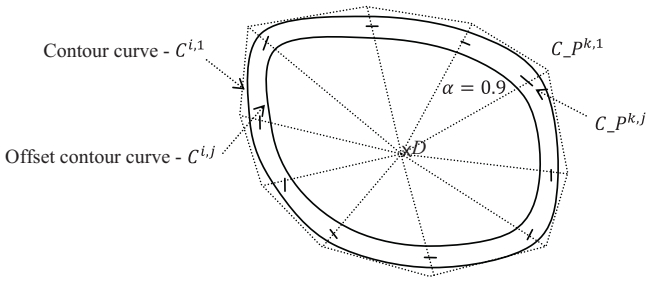


Fig. 5. An example for computing new control points of an offset contour curve.

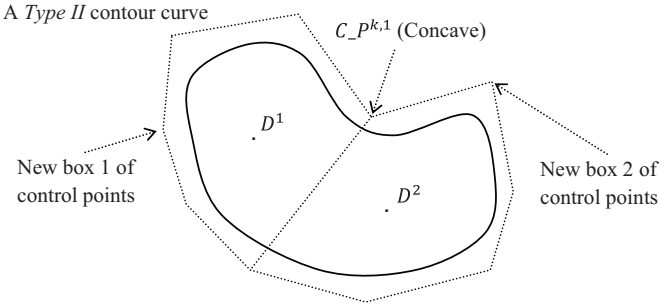


Fig. 6. An example for handling a *Type II* contour curve.

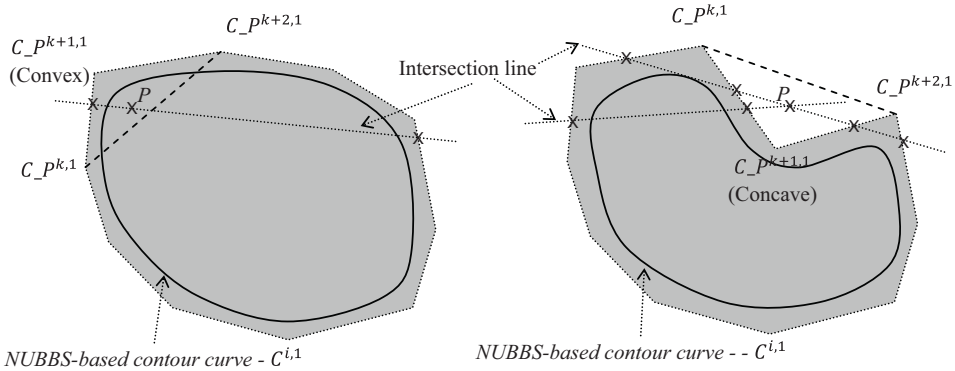
For example, in Fig. 5, a new control point $C_P^{k,j}$ of an offset contour curve is computed as follows:

$$x(C_P^{k,j}) = x(D) + 0.9(x(C_P^{k,1}) - x(D)), \quad \text{and} \quad y(C_P^{k,j}) = y(D) + 0.9(y(C_P^{k,1}) - y(D))$$

3.1.2. Offset curve and tool-path generation for a Type II contour curve

Each concave point in the box of the control points of a *Type II* contour curve will be connected with another control point in the box to separate the box as two or more boxes, in which each control point is convex. An example is shown in Fig. 6. In each box, new control points for offset contour curves are generated using Formula (3), and all the control points are then used to generate the offset contour curve.

With a series of α, control points of offset contour curves and relevant offset contour curves can be generated. Along the contour



(a) A *Type I* contour curve (b) A *Type II* contour curve

Fig. 4. Examples of two types of contour curves. (a) A *Type I* contour curve; (b) a *Type II* contour curve.

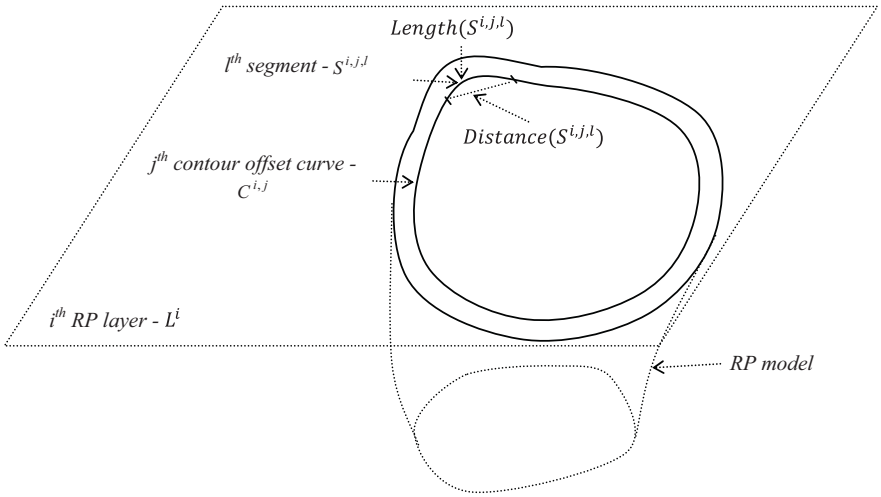


Fig. 7. Illustration of the concepts in the contour curve-based adaptive speed strategy.

curve and its series of offset contour curves, contour-based tool-paths can be interpolated.

3.2. Adaptive speed strategy of contour-based RP nozzle/print head

The RP build time for contour based tool-paths based on the boundary in the *i*th RP layer is calculated below:

$$Time(L^i) = \sum_{j=1}^{n} Time(C^{i,j}) \tag{4}$$

where *Time* represents build time; L^i represents the *i*th layer; $C^{i,j}$ represents the *j*th contour or offset curve in the *i*th layer; *n* is the total number of the contour and offset curves in the layer (including the contour curve on the boundary of the layer, and a set of offset contour curves).

A contour or offset curve is further divided as a number of segments, and each segment is defined as the same length along the curve. As thus,

$$Time(C^{i,j}) = \sum_{l=1}^{m} (Time(S^{i,j,l})) \tag{5}$$

where $S^{i,j,l}$ represents the *l*th segment in $C^{i,j}$; *m* is the total number of segments in $C^{i,j}$.

An adaptive average speed strategy of a RP nozzle/print head along a contour-based tool-path can be defined below.

$$V^{i,j,l} = \frac{Distance(S^{i,j,l})}{Length(S^{i,j,l})} V \tag{6}$$

where $V^{i,j,l}$ represents the average speed of the nozzle/print head on the segment $S^{i,j,l}$; $Length(S^{i,j,l})$ represents the curve length of the segment; $Distance(S^{i,j,l})$ represents the straight line length from the start point to the end point of the segment; *V* is a standard reference speed.

In the end, the build time spent on the segment $S^{i,j,l}$ is computed as the following.

$$Time(S^{i,j,l}) = \frac{Length(S^{i,j,l})}{V^{i,j,l}} = \frac{(Length(S^{i,j,l}))^2}{Distance(S^{i,j,l})} \frac{1}{V} \tag{7}$$

An example shown in Fig. 7 is used to illustrate the above concepts.

4. Generation of zigzag-based tool-paths

4.1. Zigzag tool-paths and adaptive strategy for slope degree optimization

Once the contour curves and related tool-paths have been generated, zigzag lines and tool-paths will be then generated to fabricate the internal area of the model. The slope of the zigzag lines are first chosen randomly and the overlapping rate between

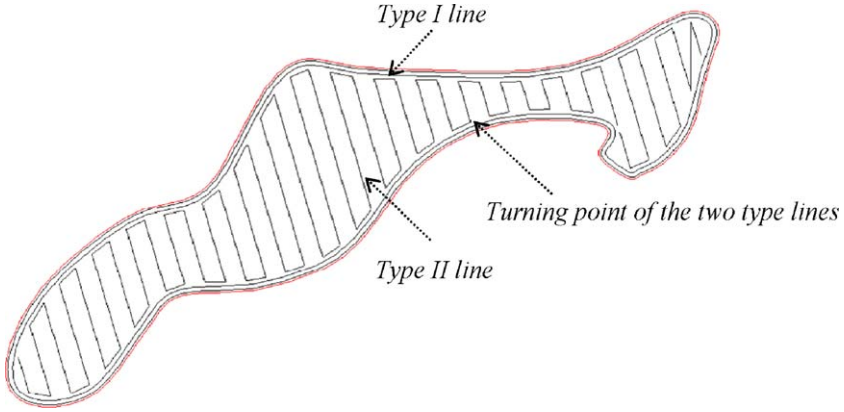


Fig. 8. Two type lines in zigzag tool-paths.

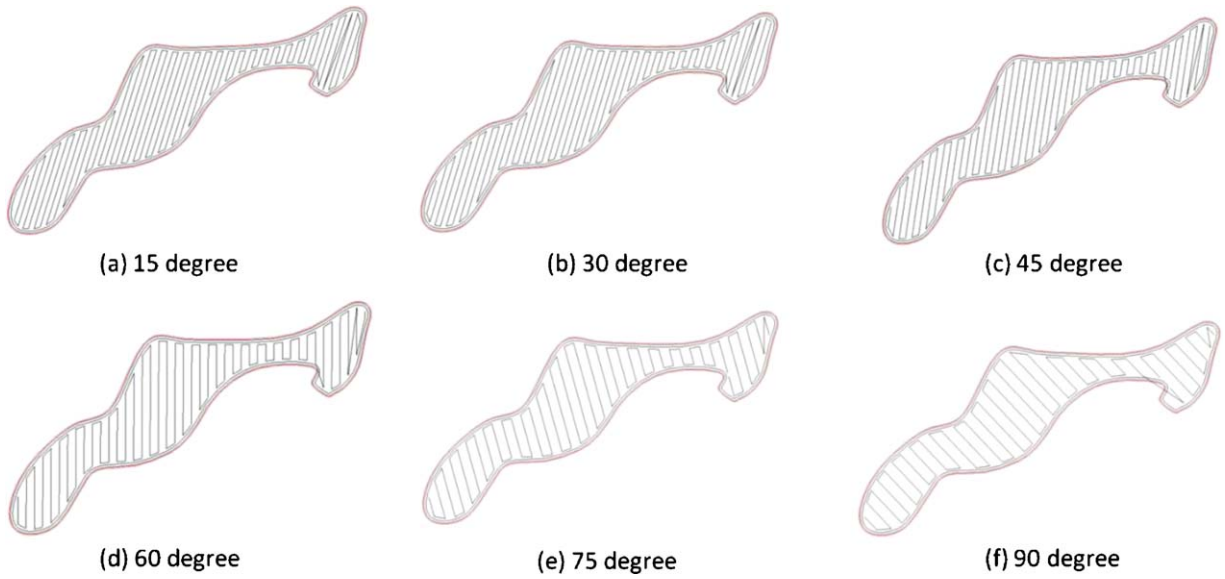


Fig. 9. Incremental degrees of slopes for the minimum build time computing. (a) 15°; (b) 30°; (c) 45°; (d) 60°; (e) 75°; (f) 90°.

two neighbour zigzag lines can be decided by users (e.g., 50%) step over. In zigzag, there are two types of lines.

Definition 3. A *Type I* zigzag line forms the main tool-paths, and a *Type II* zigzag is the connection line between two neighbouring *Type I* lines.

An example of the two type lines are shown in Fig. 8.

An adaptive strategy developed in this research for the zigzag tool-path generation is the optimization process of the slope degree of the tool-paths (i.e., the direction of the *Type I* zigzag line). The optimization objective can be modelled as:

$$Min\left(\sum_{i=1}^{n} Time(L_I^i) + \sum_{j=1}^{m} Time(L_{II}^j)\right) \tag{8}$$

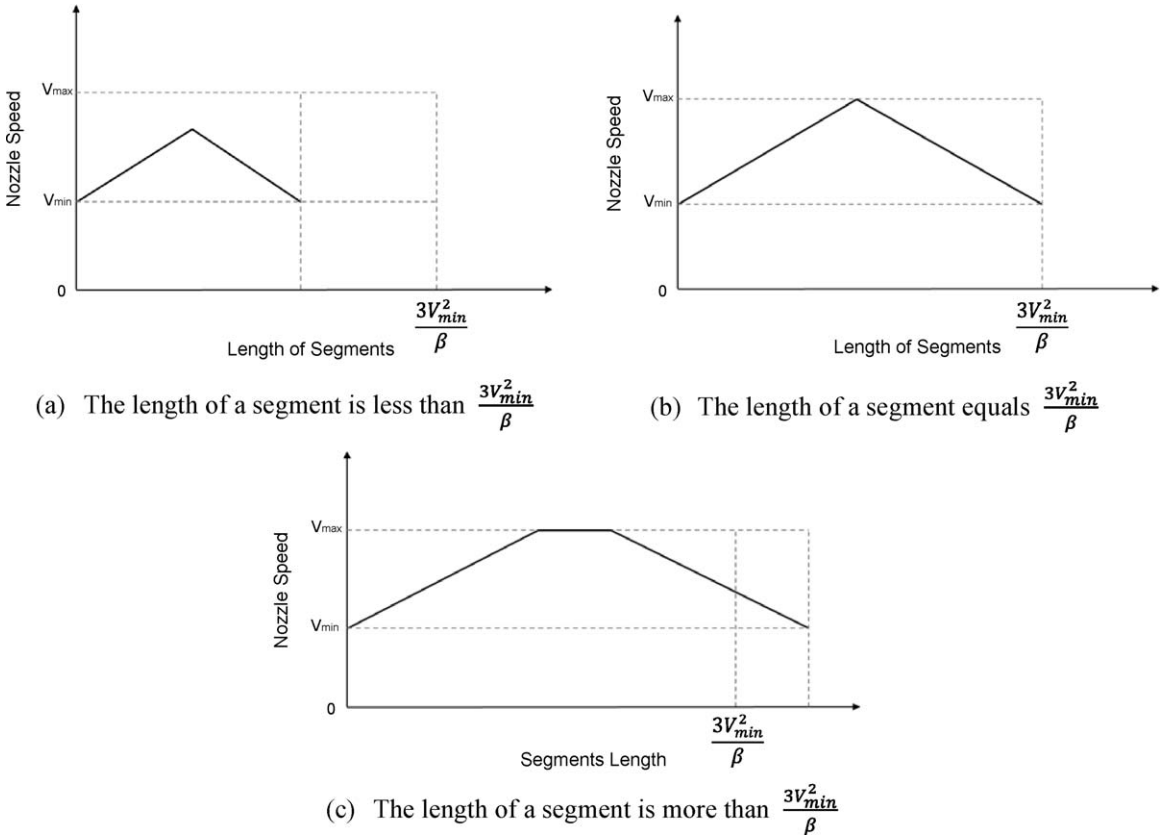


Fig. 10. Three cases of a segment and the speed of the nozzle/print head. (a) The length of a segment is less than $3V_{min}^2/\beta$; (b) the length of a segment equals $3V_{min}^2/\beta$; (c) the length of a segment is more than $3V_{min}^2/\beta$.

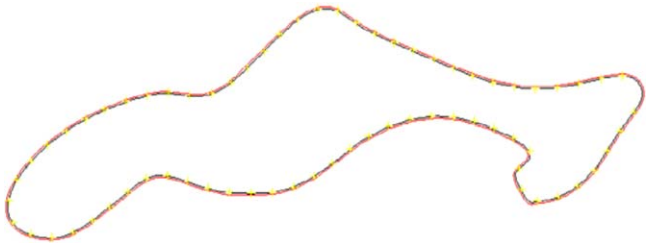


Fig. 11. Segments in two contour-based tool-paths in a layer for the ear model.

where L_I^i is the ith of the *Type I* line (n is the total number of the *Type I* lines); L_{II}^j is the jth of the *Type II* line (m is the total number of the *Type II* lines); *Time* is the build time function of both types of lines.

The variable for the optimization objective is the slope of the zigzag direction ((i.e., the direction of the *Type I* zigzag line), which can be rotated in a scope of [0°, 180°]. In the example of Fig. 9, the slope degree can be incrementally changed, and the minimum build time can be obtained through the comparison of the computation results for all the degrees.

4.2. Adaptive strategy for RP nozzle/print head speed

Each *Type I* or *Type II* line consists of a number of segments. The build time of these segments can be calculated in the following process:

$$Time(L_I^i) = \sum_{k=1}^{n} Time(S_I^{i,k}) \tag{9}$$

$$Time(L_{II}^j) = \sum_{k=1}^{m} Time(S_{II}^{j,k}) \tag{10}$$

where *Time* is the build time function; L_I^i represents the ith *Type I* line, and $S_I^{i,k}$ represents the kth segment in the line (n is the total number of the segment in the line); L_{II}^j represents the jth *Type II* line, and $S_{II}^{j,k}$ represents the kth segment in the line (m is the total number of the segment in the line).

In the following, $Time(S_I^{i,k})$ and $Time(S_{II}^{j,k})$ are computed further:

$$Time(S_I^{i,k}) = \frac{Length(S_I^{i,k})}{Velocity(S_I^{i,k})} \tag{11}$$

$$Time(S_{II}^{j,k}) = \frac{Length(S_{II}^{j,k})}{Velocity(S_{II}^{j,k})} \tag{12}$$

where *Length* is the length function; *Velocity* is the speed function of the RP nozzles/print heads along the tool-paths.

An adaptive speed to optimize the movement of the RP nozzles/print heads along zigzag tool-paths has been developed. That is, for a *Type I* or *Type II* line, a nozzle/print head starts from a minimum speed, accelerates afterwards towards a maximum speed, and then decelerates to the minimum speed at the end of the entire line. The maximum speed, minimum speed, acceleration and deceleration are determined by the specification of the RP machine which is being used. The design is aimed to improve the efficiency of the RP process by accelerating or decelerating the nozzle/print head in the different stages of zigzag. As thus, the following assumptions are made:

(1) The speed of a nozzle/print head from all the turning points of the two type lines is the minimum speed (represented as V_{min});
(2) The maximum speed that the nozzle/print head can achieve is V_{max}. The minimum speed is V_{min}. They are determined by the used RP machine. Here, V_{max} is set twice of V_{min};
(3) The speed of the nozzle/print head in the zigzag tool-paths is either uniformly accelerated or deceleration with an acceleration as β.

Based on that, the adaptive strategy can be further represented as follows.

$$V_{\max} = 2V_{\min} \tag{13}$$

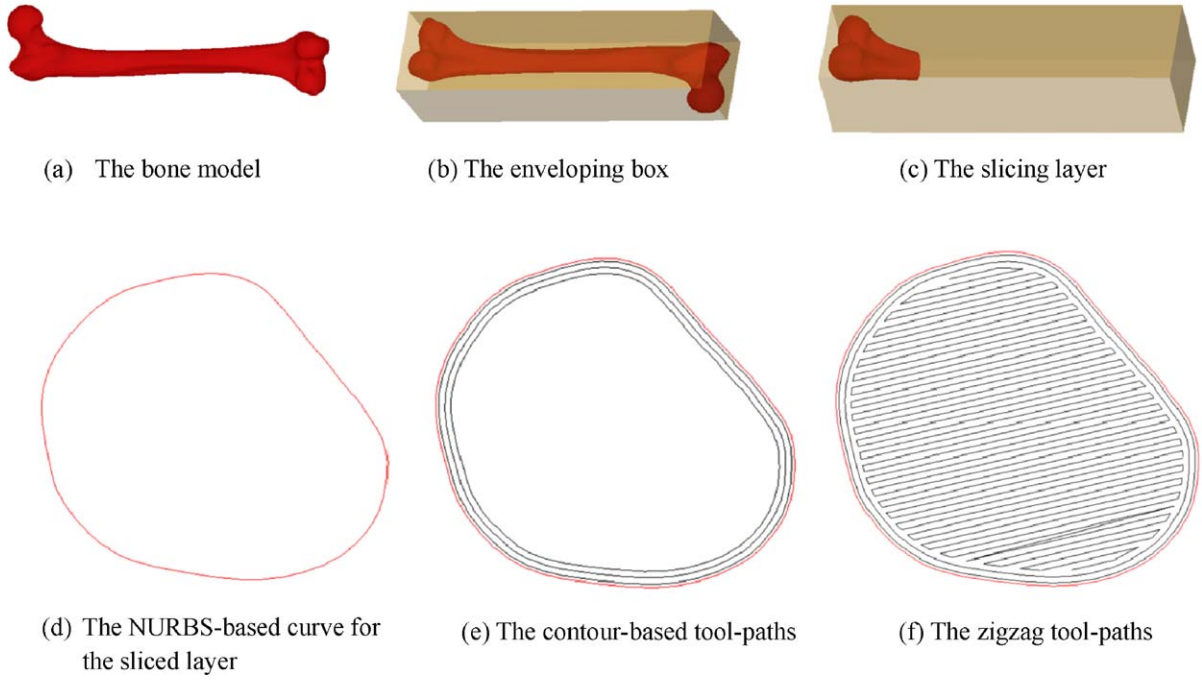


Fig. 12. The tool-path generation for the bone model. (a) The bone model; (b) the enveloping box; (c) the slicing layer; (d) the NURBS-based curve for the sliced layer; (e) the contour-based tool-paths; (f) the zigzag tool-paths.

$$V_{max} = V_{min} + \beta t \tag{14}$$

where t is the time used to speed up the nozzle/print head from the minimum speed to the maximum speed.

From Formulas (13) and (14), t can be deduced as:

$$t = \frac{V_{min}}{\beta} \tag{15}$$

On the other hand,

$$Length = \frac{1}{2}\beta t^2 + V_{min} t \tag{16}$$

The following will be obtained if Formulas (15) and (16) are combined:

$$Length = \frac{3V_{min}^2}{2\beta} \tag{17}$$

Based on above analysis, there are three cases for a nozzle/print head to move on a segment:

(1) If the length of the segment is less than $3V_{min}^2/\beta$, the nozzle/print head's speed could not reach the maximum speed V_{max} (shown in Fig. 10(a));
(2) If the length of the segment is equals to $3V_{min}^2/\beta$, the nozzle/print head's speed just reach the maximum speed V_{max} in the middle of the segment (shown in Fig. 10(b));
(3) If the length of the segment is more than $3V_{min}^2/\beta$, the nozzle/print head's speed reach the maximum speed V_{max} before the middle of the segment (shown in Fig. 10(c)).

To summarize Formulas (11)–(17), the time of a *Type I* or *Type II* line of zigzag will be computed as follows:

$$Time(L_I^{i,k}) = \begin{cases} \dfrac{2V_{min}}{\beta} & \text{if } Length(L_I^{i,k}) = \dfrac{3V_{min}^2}{\beta} \\ \dfrac{-2V_{min}}{\beta} + \sqrt{\left(\dfrac{2V_{min}}{\beta}\right)^2 + \dfrac{4Length(L_I^{i,k})}{\beta}} & \text{if } 0 < Length(L_I^{i,k}) < \dfrac{3V_{min}^2}{\beta} \\ \dfrac{2V_{min}}{\beta} + \dfrac{\alpha \times Length(L_I^{i,k}) - 3V_{min}^2}{2\beta \times V_{min}} & \text{if } \dfrac{3V_{min}^2}{\beta} < Length(L_I^{i,k}) \end{cases} \tag{18}$$

$$Time(L_{II}^{j,k}) = \begin{cases} \dfrac{2V_{min}}{\beta} & \text{if } Length(L_{II}^{j,k}) = \dfrac{3V_{min}^2}{\beta} \\ \dfrac{-2V_{min}}{\beta} + \sqrt{\left(\dfrac{2V_{min}}{\beta}\right)^2 + \dfrac{4Length(L_{II}^{j,k})}{\beta}} & \text{if } 0 < Length(L_{II}^{j,k}) < \dfrac{3V_{min}^2}{\beta} \\ \dfrac{2V_{min}}{\beta} + \dfrac{\alpha \times Length(L_{II}^{j,k}) - 3V_{min}^2}{2\beta \times V_{min}} & \text{if } \dfrac{3V_{min}^2}{\beta} < Length(L_{II}^{j,k}) \end{cases} \tag{19}$$

5. Implementation and case studies

The research presented in this paper has been implemented using the C++ programming language in the Open CASCADE platform, which is an open-source CAD kernel system [26]. A number of complex product model case studies have been tested for feasibility studies and it has proved that the effectiveness and reliability of the developed research are satisfactory. Two case studies are presented below for demonstration.

5.1. Case study 1 – an ear model

An ear model has been used for the tool-path generation of RP using the methodology developed in this research. The process and geometrical results have already been presented in Figs. 2, 8 and 9. The average speeds and build time for each segment of two contour-based tool-paths in a layer for the ear model (the segments are shown in Fig. 11) are computed. The results are displayed in Table 1.

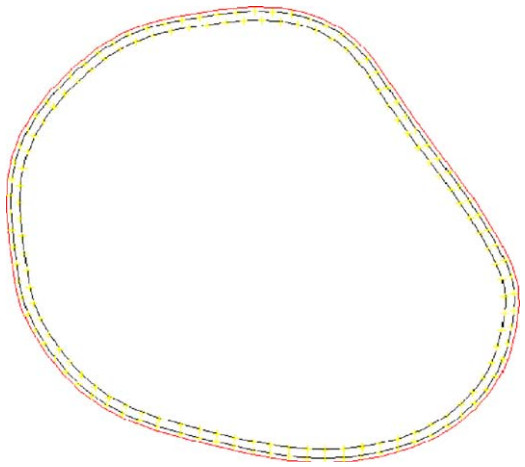


Fig. 13. Segments in two contour-based tool-paths in a layer for the bone model.

For the zigzag tool-paths, different slope degrees were used for the layer (shown in Fig. 9). The minimum speed of the nozzle/print head is defined as 5 mm/s, and the acceleration of the nozzle/print head is **25 mm/s²** (V_{min} = 5 mm/s; a = 25 mm/s²). Table 2 shows the results of each build time with an incremental 15° for the slope from 0° until 180°. It can be observed that in 90° the build time is the shortest. Totally build time for the layer of the ear model = 10.217 + 10.117 + 20.533 = **40.867 s** (contour-based tool-paths (two curves in this case) + zigzag tool-paths). All results were obtained from a computer system with a Pentium Dual-Core CPU 2.10 GHz and 2GB RAM. The amount of time taken for the computation of different slope degrees for the layer is 2.13 s while the average reduction time of the RP fabrication process by using this optimization strategy is 16 s for a single layer. A model could consist of hundreds or thousands of layers. It will be much less if the computer is more powerful.

5.2. Case study 2 – a bone model

A bone model has also been used to validate the research. The slicing process for the NURBS-based contour curve generation, and the generation of contour-based tool-paths and zigzag tool-paths are shown in Fig. 12, the segments of the contour curve are shown in Fig. 13. The results of the nozzle/print head speed in the contour-based tool-paths and zigzag tool-paths are shown in Table 3 and Table 4 respectively.

The conditions were set the same as that for the above ear model. The results of each build time for the bone model from 0 degree until 180° are shown in Table 4. We can find that the build time

Table 1
Nozzle/print head speed of the contour-based tool-paths for the ear model.

	Segment no.																		
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
Contour tool-path one (total build time for contour tool-path one = 10.217 s)																			
Curve length	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500
Length (mm)	4.002	3.941	3.992	3.966	3.710	3.990	3.803	4.020	4.014	3.650	3.868	3.985	3.836	4.019	4.006	3.916	3.904	2.766	3.127
Average speed	8.895	8.755	8.870	8.815	8.245	8.865	8.450	8.935	8.920	8.105	8.595	8.855	8.525	8.930	8.900	8.700	8.675	6.145	6.950
Time (s)	0.506	0.514	0.507	0.510	0.546	0.508	0.533	0.504	0.504	0.555	0.524	0.508	0.528	0.504	0.056	0.517	0.519	0.732	0.647
Contour tool-path two (total build time for contour tool-path two = 10.117 s)																			
Curve length	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500
Length (mm)	4.018	3.870	3.976	3.927	3.922	3.964	3.710	4.094	4.101	3.754	4.080	3.802	3.912	4.115	4.029	3.829	3.822	2.504	3.672
Average speed	8.930	8.600	8.835	8.725	8.715	8.810	8.245	9.100	9.115	8.345	9.065	8.450	8.690	9.145	8.955	8.510	8.495	5.565	8.160
Time (s)	0.504	0.523	0.509	0.516	0.516	0.511	0.546	0.495	0.494	0.539	0.496	0.533	0.518	0.492	0.503	0.529	0.530	0.809	0.551

Table 2
Comparison of computational results of different slope degrees of zigzag tool-paths for the ear model.

Slope degree	0°	15°	30°	45°	60°	75°	**90°**	105°	120°	135°	150°	165°	180°
Curve length (mm)	364.716	351.032	327.795	292.046	246.380	198.212	**181.420**	198.102	254.594	341.825	385.597	416.420	401.256
Build time (s)	40.148	39.611	36.911	32.803	27.596	22.636	**20.533**	21.774	26.642	36.860	38.659	42.929	40.458

Table 3
Nozzle speed of the contour offset tool-path with bone model.

	Segment no.																								
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
Contour tool-path one (total build time for contour tool-path one = 12.469 s)																									
Curve length	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500
Length (mm)	4.071	4.078	4.069	4.064	4.032	4.084	4.078	4.055	4.085	4.048	4.051	4.099	4.098	4.017	3.981	4.020	4.031	4.102	4.074	4.098	4.089	4.068	4.059	4.038	4.013
Average speed	9.045	9.065	9.045	9.030	8.960	9.075	9.065	9.010	9.080	8.995	9.000	9.110	9.110	8.925	8.845	8.935	8.960	9.115	9.055	9.110	9.085	9.040	9.020	8.970	8.920
Time (s)	0.498	0.496	0.498	0.498	0.502	0.496	0.496	0.499	0.496	0.500	0.500	0.494	0.494	0.504	0.509	0.504	0.502	0.494	0.497	0.494	0.495	0.498	0.499	0.502	0.504

	Segment no.																							
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
Contour tool-path two (total build time for contour tool-path two = 11.553 s)																								
Curve length	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500	4.500
Length (mm)	4.253	4.275	4.265	4.166	4.148	4.326	4.242	4.220	4.298	4.101	4.275	4.442	4.215	3.937	4.017	4.079	4.315	4.331	4.353	4.340	4.248	4.207	4.132	4.040
Average speed	9.450	9.500	9.475	9.260	9.220	9.615	9.430	9.386	9.550	9.115	9.500	9.870	9.365	8.750	8.925	9.065	9.585	9.620	9.675	9.645	9.440	9.350	9.200	8.980
Time (s)	0.476	0.470	0.475	0.486	0.488	0.468	0.477	0.479	0.471	0.494	0.474	0.456	0.481	0.514	0.504	0.496	0.496	0.468	0.465	0.467	0.477	0.481	0.489	0.501

Table 4
Comparison of computational results of different slope degrees of zigzag tool-paths for the bone model.

Slope degree	**0**°	15°	30°	45°	60°	75°	90°	105°	120°	135°	150°	165°	180°
Curve length (mm)	**969.030**	988.186	1035.690	1168.120	1475.270	1552.880	1742.316	1556.310	1485.180	1171.090	1050.640	1005.730	987.396
Build time (s)	**99.683**	100.241	106.920	120.648	158.258	163.764	184.533	164.103	151.349	121.025	108.108	100.389	98.985

in 0° is the shortest. Totally build time for the layer of the bone model = 12.469 + 11.553 + 99.683 = **123.705 s**. The amount of time taken for computation of different slope degrees for the layer is 2.51 s while the average reduction time of the RP fabrication process by using this optimization strategy is 46 s for a single layer.

6. Conclusions

This paper focuses on improving surface accuracy and reducing build time for complex CAD model fabrication in RP systems. A series of optimization algorithms and strategies have been developed to optimize the tool-path generation in RP process planning. The developed algorithms and strategies provide a novel and systematic means to improve the surface quality and overall fabrication efficiency. The major features of the research are in the following aspects.

The geometrical accuracy of original CAD models are maintained through slicing on the CAD models to keep high-fidelity information and introducing NURBS to represent the contours of sliced layers. It establishes a good original data source for further RP tool-path generation.

Meanwhile, the developed mixed tool-path generation algorithms can effectively balance the requirements on RP surface quality and fabrication efficiency. It is an adaptive process to address various complex product models. Complex biomedical models have been used to justify the effectiveness and robustness of the algorithms.

The developed build time analysis mathematical models and adaptive speed strategies for RP nozzles/print heads can be used to optimize build time towards a minimum target.

References

[1] Kruth JP, Levy G, Klocke F, Childs THC. Consolidation phenomena in laser and powder-bed based layered manufacturing. Annals of CIRP 2007;56:730–59.
[2] Bourell DL, Leu MC, Rosen DW. Roadmap for additive manufacturing: identifying the future of freeform processing. Austin, TX: The University of Texas at Austin; 2009.
[3] Sabourin E, Houser SA, Bohn JH. Accurate exterior, fast interior layered manufacturing. Rapid Prototyping Journal 1997;3:44–52.
[4] Ren L, Sparks T, Ruan JZ, Liou F. Process planning strategies for solid freeform fabrication of metal parts. Journal of Manufacturing Systems 2008;27:158–65.
[5] Marsan A, Dutta D. A survey of process planning techniques for layered manufacturing. In: Proceedings of the 1997 ASME design automation conference. 1997.
[6] Thompson DC. The optimization of part orientation for solid freeform manufacture. Master of science thesis. The University of Texas at Austin; 1995.
[7] Magics RP software. Information on http://www.materialise.com/materialise/view/en/2408555-Magics.html [last access on 12th May 2011].
[8] VisCAM RP software. Information on http://www.marcam.de/cms/index.85.en.html [last access on 12th May 2011].
[9] Kulkarni P, Marsan A, Dutta A. A review of process planning techniques in layered manufacturing. Rapid Prototyping Journal 2000;6(1):18–35.
[10] Chen X, Wang C, Ye X, Xiao Y, Huang S. Direct slicing from PowerSHAPE models for rapid prototyping. International Journal of Advanced Manufacturing Technology 2001;17:543–7.
[11] Shi Y, Chen X, Cai D, Huang S. Application software system based on direct slicing for rapid prototyping. International Journal of Product Research 2004;42:2227–42.
[12] Rianmora S, Koomsap P. Recommended slicing positions for adaptive direct slicing by image processing technique. International Journal of Advanced Manufacturing Technology 2010;46:1021–33.
[13] Ma W, But WC, He P. NURBS-based adaptive slicing for efficient rapid prototyping. Computer-Aided Design 2004;36:1309–25.
[14] Liu Z, Wang L, Lu B. Integrating cross-sectional imaging based reverse engineering with rapid prototyping. Computers in Industry 2006;57:131–40.
[15] Misra D, Sundararajan V, Wright PK. Zig-zag tool path generation for sculptured surface finishing. Dimacs Series in Discrete Mathematics and Theoretical Computer Science 2005;67:265–80.
[16] Chang WR. CAD/CAM for the selective laser sintering process. MS thesis. University of Texas, Austin, USA; 1997.
[17] Dwivedi R, Kovacevic R. Automated torch path planning using polygon subdivision for solid freeform fabrication based on welding. Journal of Manufacturing Systems 2004;23(4):278–91.
[18] Yang Y, Loh HT, Fuh FYH, Wang YG. Equidistant path generation for improving scanning efficiency in layered manufacturing. Rapid Prototyping Journal 2002;8(1):30–7.
[19] Genesan M, Fadel G. Hollowing rapid prototyping parts using offsetting techniques. In: Proceedings of the 5th international conference on rapid prototyping. 1994. p. 241–51.
[20] Bertoldi M, Yardimci MA, Pistor CM, Guceri SI. Domain decomposition and space filling curves in toolpath planning and generation. In: Proceedings of the 1998 solid freeform fabrication symposium. 1998. p. 267–74.
[21] Chen CC, Sullivan PA. Predicting total build-time and the resultant cure depth of the 3D stereolithography process. Rapid Prototyping Journal 1996;2(4):27–40.
[22] Han W, Jafari MA, Seyed K. Process speeding up via deposition planning in fused deposition-based layered manufacturing processes. Rapid Prototyping Journal 2003;9(4):212–8.
[23] Wah PK, Murty KG, Joneja A, Chiu LC. Toolpath optimization in layered manufacturing. IIE TRANSACTIONS 2002;34(4):335–47.
[24] Castelino K, D'Souza R, Wright PK. Toolpath optimization for minimizing airtime during machining. Journal of Manufacturing Systems 1999;22(3):173–80.
[25] Ma YL, Hewitt WT. Point inversion and projection for NURBS curve and surface: control polygon approach. Computer Aided Geometrical Design 2003;20(2):79–99.
[26] Open CASCADE. Information on http://www.opencascade.org/ [last access on 12th May 2011].